

On the Alignment, Robustness, and Generalizability of Multimodal Learning

Jielin Qiu

CMU-CS-24-101

April, 2024

Computer Science Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA, USA

Thesis Committee:

Christos Faloutsos (Co-chair)

Lei Li (Co-chair)

Yonatan Bisk

William Wang (University of California, Santa Barbara)

*Submitted in partial fulfillment of the requirements
for the Degree of Doctor of Philosophy*

Copyright © 2024 Jielin Qiu

This research was sponsored in part by CMU CSD fellowships, the Defense Advanced Research Projects Agency (DARPA) ADAPTER program, Adobe Research Gift Funding, Allegheny Health Network, Mario Lemieux Center for Innovation and Research in EP, and Cleveland Clinic. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of any sponsoring institution, the U.S. government, or any other entity. Creative Commons License: CC-BY-NC-SA.

Keywords: multimodal learning, semantic alignment, multimodal robustness, generalization, cross-domain alignment

Abstract

Multimodal intelligence, where AI systems can process and integrate information from multiple modalities, such as text, visual, audio, etc., has emerged as a key concept in today’s data-driven era. This cross-modal approach finds diverse applications and transformative potential across industries. By fusing heterogeneous data streams, multimodal AI generates representations more akin to human-like intelligence than traditional unimodal techniques.

In this thesis, we aim to advance the field of multimodal intelligence by focusing on three crucial dimensions: multimodal alignment, robustness, and generalizability. By introducing new approaches and methods, we aim to improve the performance, robustness, and interpretability of multimodal models in practical applications. In this thesis, we address these critical questions: (1) How do we explore the inner semantic alignment between different types of data? How can the learned alignment help advance multimodal applications? (2) How robust are the multimodal models? How can we improve the models’ robustness in real-world applications? (3) How do we generalize the knowledge of one learned domain to another unlearned domain?

This thesis makes contributions to all three technical challenges. We start with a contribution of learning cross-modal semantic alignment, where we explore establishing rich connections between language and image/video data, with a focus on the multimodal summarization task. By aligning the semantic content of language with visual elements, the resulting models can possess a more nuanced understanding of the underlying concepts. We delve into the application of Optimal Transport-based approaches to learn cross-domain alignment, enabling models to provide interpretable explanations of their multimodal reasoning process.

For the next contribution, we develop comprehensive evaluation metrics and methodologies to assess the robustness of multimodal models. By simulating distribution shifts and measuring the model’s performance under different scenarios, we can gain a deeper understanding of the model’s adaptability and identify potential vulnerabilities. We also adopt Optimal Transport to improve the model’s robustness performance through data augmentation via Wasserstein Geodesic perturbation.

The third contribution revolves around the generalizability of multimodal systems, with an emphasis on the interactive domain and the healthcare domain. In the interactive domain, we develop new learning paradigms for learning executable robotic policy plans from visual observations by incorporating latent language encoding. We also use retrieval augmentation to make the vision-language models capable of recognizing and providing knowledgeable answers in real-world entity-centric VQA. In the healthcare domain, we bridge the gap by transferring the knowledge of LLMs to clinical ECG and EEG. In addition, we design retrieval systems that can automatically match the clinical healthcare signal to the most similar records in the database. This functionality can significantly aid in diagnosing diseases and reduce physicians’ workload.

In essence, this thesis seeks to propel the field of multimodal AI forward by enhancing alignment, robustness, and generalizability, thus paving the way for more sophisticated and efficient multimodal AI systems.

Acknowledgments

These five years at CMU have been truly an amazing journey. I am incredibly grateful for the opportunity to express my profound appreciation to my advisors, Prof. Lei Li and Prof. Christos Faloutsos, for their unwavering support and invaluable mentorship throughout my educational and personal development journey. Their guidance during times of uncertainty and challenge was not just professional but also deeply personal, fostering a nurturing environment for my growth. Lei has been an exceptional advisor, teaching me the intricacies of critical thinking and research. His expertise and hands-on approach in guiding me through complex projects have been instrumental in honing my skills. Lei's ability to delve deep into impactful work and illuminate the path towards innovative solutions has had a lasting impact on my approach to research and problem-solving. Christos has been a pillar of support and wisdom. Whether I needed advice on academic matters or help in preparing for presentations, Christos was always there with his insightful guidance and unwavering support. His readiness to assist at any moment and his profound understanding of the academic landscape has greatly contributed to my confidence and preparedness in various academic endeavors. Their combined mentorship went beyond mere academic instruction; it was holistic guidance that encompassed all aspects of personal and professional development. Their consistent encouragement and faith in my potential provided me with the strength and resilience to navigate through challenging periods in my academic career. Their belief in my abilities served as a powerful catalyst, propelling me towards achieving my goals and surpassing expectations. Moreover, the kind and amiable personalities of Lei and Christos, combined with their meticulous and thoughtful approach to mentorship, have been a continuous source of inspiration. They have set an exemplary model for me to emulate in my future academic and professional engagements. The impact of their mentorship extends beyond the confines of academia; it has shaped my character and approach to life's challenges.

I am profoundly thankful for the valuable contributions of my thesis committee members, Prof. Yonatan Bisk and Prof. William Wang. The insightful feedback and comprehensive perspectives provided by Yonatan and William have not only enhanced the quality of my work but also encouraged me to explore more complex and broader intellectual areas. My association with Yonatan began when I was a teaching assistant in his course. He has imparted extensive knowledge and demonstrated effective ways of collaborating with others through his cheerful demeanor and genuine kindness. These qualities have consistently brought a positive atmosphere to our interactions, greatly enriching my learning experience. William has been exceptional in identifying the core significance of my work and offering invaluable advice for its enhancement. His proficiency in research and meticulous attention to detail have played a crucial role in my development as a researcher. Their support has not only aided me in achieving academic success but also in developing a deeper understanding and appreciation of the research process. Their influence extends beyond the confines of my thesis, leaving a lasting impact on my approach to academic and professional challenges.

I am immensely grateful for the mentorship and support I received from several academic mentors during my PhD studies. Each of them has played a unique and

invaluable role in my development as a researcher and individual. I owe a special debt of gratitude to Prof. Bo Li from the University of Chicago. Under Bo's guidance for two years, I have received consistent, insightful advice and suggestions that significantly improved my work. Bo's vast knowledge across various fields and ability to pinpoint the most critical aspects of a problem have been incredibly influential in my academic growth. I am also deeply appreciative of Prof. Doug Weber's generosity. His funding support for two years and his ongoing encouragement were pivotal in facilitating my research endeavors. Prof. Daniel Fried deserves special mention for his detailed and constructive feedback on my work. Our discussions were always enlightening, and I learned a great deal from his guidance on crafting high-quality research papers. Daniel's meticulous approach to academic discourse has been a great source of learning for me. My gratitude extends to Prof. Jun-Yan Zhu as well, with whom I formed both a mentorship and a friendship. Since meeting Jun-Yan as a teaching assistant in his course, I have gained immense knowledge. His advice, covering both academic and personal aspects, especially during uncertain times in my life, has been incredibly valuable. I am also thankful for the mentorship of Dr. Emerson Liu from Allegheny Health Network and Dr. Michael Rosenberg from the University of Colorado Cardiovascular Institute. Their teachings in the clinical domain laid the foundation for my work in healthcare, providing me with the essential knowledge needed to bridge the gap between technical and clinical aspects. The opportunity to learn from Prof. XuanLong Nguyen has been another highlight of my PhD journey. XuanLong exemplified what it means to be a dedicated and decent researcher, setting a standard for me to aspire to in my academic career. I also greatly appreciate Prof. Tianqi Chen and Prof. Nancy Pollard for helping me with my CSD Speaking Skills. Lastly, my heartfelt thanks go to Prof. Fei Fang. Her support and guidance during challenging times have been extraordinary. Fei's positive attitude and bright smile have a way of making complex problems seem more manageable, providing me with a perspective that often led to better solutions than I could have imagined. Her influence has been a beacon of optimism in my academic journey. I am also fortunate to learn from senior researchers, including Dr. Xiuming Zhang from MIT, Dr. Alex Smola from Boson AI, Dr. Tristan Naumann, Dr. Ozan Oktay from Microsoft, Prof. Weina Wang, Prof. Ding Zhao, Prof. Jeff Schneider, Prof. Leila Wehbe, Prof. Zico Kolter, Prof. Changliu Liu, Prof. David Held, Prof. Leman Akoglu, Prof. Steven Brookes from CMU.

I am profoundly grateful for the mentorship and support I received from numerous esteemed mentors during my internships, each of whom has significantly contributed to my professional and personal growth. I extend my sincere thanks to Dr. Yi Zhu, who has been an outstanding mentor since my time interning at AWS. Yi's insights and guidance were crucial in nurturing my development during crucial stages of my academic journey. Alongside Dr. Xingjian Shi, also at AWS, I was fortunate to gain extensive knowledge and research skills and experience their warm-hearted approach to mentorship. My gratitude also goes to Dr. Hailin Jin, whose exceptional support over the past three years has been instrumental in my development. As my manager during my internship at Adobe, Hailin offered me invaluable guidance and the opportunity to delve into a new research topic that later contributed to a patent. His availability and willingness to assist even after my internship are deeply appreciated. The internship

at Meta was another enriching experience, where I learned immensely from Dr. Luna Dong, Dr. Shane Moon, Dr. Zhaojiang Lin, and Dr. Andrea Madotto. The team at Meta provided an environment that exceeded my expectations, offering a unique and valuable learning experience. Across my five internships, I have been privileged to work with and learn from a group of exceptional mentors, including Dr. Hailin Jin, Dr. Zhaowen Wang, Dr. Trung Bui, Dr. Franck Deroncourt from Adobe; Dr. Yi Zhu, Dr. Xingjian Shi, Dr. Zhiqiang Tang, Dr. Mu Li from Amazon Web Service; Dr. Jianfeng Wang, Linjie Li, Dr. Zhengyuan Yang, Dr. Lijuan Wang from Microsoft; Dr. Shane Moon, Dr. Andrea Madotto, Dr. Zhaojiang Lin, Dr. Luna Dong, Dr. Ethan Xu, Dr. Paul Crook, Babak Damavandi from Meta; and Shangbang Long from Google. The opportunity to work alongside and learn from these brilliant individuals has been an absolute pleasure and an enriching experience.

I am filled with heartfelt gratitude for my collaborators and friends, whose presence has greatly enriched both my academic journey and personal life. Their insightful discussions, unwavering support, and shared moments of joy have been invaluable. Special thanks to my friend Jiacheng Zhu, whose reliability and inspiring work ethic have continually motivated me. To Mengdi Xu, with whom I have shared not only life experiences but also cultivated research ideas and sought advice, my unending gratitude. I am grateful for the climbing adventures and great moments shared with Zuxin Liu and Rui Chen. These experiences have added a unique and enjoyable dimension to my life. My profound appreciation equally goes to close friends Peide Huang, Yiwen Dong, Hanjiang Hu, Yuyou Zhang, Wenhao Ding, Miao Li, Haohong Lin, Laixi Shi, Yaru Niu, William Han, Shiqi Liu, Zhepeng Cen, Yihang Yao, Changyi Lin, Diana Gomez – with whom I have forged a bond akin to family, supporting each other through thick and thin. I also extend thanks to my labmates Zhenqiao Song, Danqing Wang, Xuandong Zhao, Siqui Ouyang, Kexun Zhang, and Wenda Xu for their support and camaraderie. Additionally, I cherish the friendships formed within the CSD community, including with Yi Zhou, Mingjie Sun, Shuqi Dai, Chen Dan, Dorian Chan, Long Pham, Minji Yoon, and my office mates Lucio Dery, Asher Trockman, Praneeth Kacham, and Suhas Jayaram Subramanya. I am equally thankful for the companionship of friends like Scarlett Zhang, Thomas Ning, Weiye Zhao, Tian Lu, Chengyuan Zhang, Jingqi Zhang, Zilin Si, Yeeho Song, Helen Jiang. Furthermore, I have been fortunate to have close friends and collaborators outside CMU, such as Cindy Wu from Princeton, Bo He, Jun Wang from UMD, Zhuolin Yang from UIUC, Zijie Huang from UCLA, Aritra Guha from the University of Michigan, Makiya Nakashima, Jaehyun Lee, Debbie Kwon, David Chen from Cleveland Clinic, and Florian Wenze from Amazon. The shared experiences, laughter, and support from all these wonderful individuals have made my journey not only successful but also joyous and fulfilling. Their companionship has been a source of immense joy and a key factor in my personal and professional growth.

I want to express my deep gratitude to the Computer Science Department for their support and encouragement. A heartfelt thank you to Prof. Karl Crary and Prof. Srinivasan Seshan for their steadfast belief in me and their generous assistance, especially during challenging periods. Their unwavering support has been a fundamental pillar in my academic journey. I am also profoundly thankful to Deb Cavlovich and Matthew Stewart for their invaluable and continuous support, which has greatly simpli-

fied many complex tasks. Their assistance has been crucial in navigating the intricacies of academic processes. Additionally, my appreciation goes to Nick Hernandez, Jenn Landefeld, Amanda Hornick, and Tracy Farbacher for their timely support and commitment. Their efforts have played a significant role in fostering a smooth and supportive academic environment, making my journey both successful and enjoyable.

In closing, my most profound and heartfelt thanks are reserved for my entire family, whose constant love and support have been the foundation of everything I have accomplished. To my parents, Guoming and Hua, my gratitude is immense and truly without limits. Their endless love, unwavering support, and consistent encouragement have been the guiding forces in my life. Their steadfast belief in me has illuminated every step of my journey. To my beloved wife, Zhenni, your unwavering encouragement and profound faith in me have served as the cornerstone of my strength. Your presence in my life has not merely illuminated my journey but has consistently been a wellspring of inspiration and solace, enriching every step I take. Special thanks to my cat, who has been my unwavering buddy, offering comfort and companionship through every milestone and tearful moment. This journey, brimming with triumphs and challenges, would have been unimaginable without your unwavering love and support. This achievement reflects not only my efforts but equally your sacrifices and steadfast backing. I am deeply grateful for your constant support and the numerous sacrifices you have made. To all of you, my gratitude knows no bounds.

To Guoming, Hua, and Zhenni

Contents

1	Introduction	1
1.1	Motivation and Challenges	2
1.2	Thesis Overview	5
1.3	Summary of Contributions	7
2	Background and Preliminary	11
2.1	Background	11
2.2	Preliminary	12
I	Learning Cross-modal Semantic Alignment	15
3	Multimodal Summarization via Cross-domain Alignment	16
3.1	Introduction	16
3.2	Related Work	18
3.3	Proposed Method	18
3.3.1	Video Temporal Segmentation	19
3.3.2	Textual Segmentation	20
3.3.3	Visual Summarization	20
3.3.4	Textual Summarization	21
3.3.5	Cross-Domain Alignment via OT	21
3.3.6	Multimodal Summaries	22
3.4	Datasets and Baselines	22
3.5	Experiments	23
3.6	Conclusion	27
4	Unsupervised Multimodal Temporal Segmentation of Long Livestream Videos	28
4.1	Introduction	28
4.2	Related Work	30
4.3	MultiLive Dataset	31
4.4	Proposed Method	34
4.5	Experiments and Results	39
4.6	Conclusion	42

5	MMSum: A Dataset for Multimodal Summarization	43
5.1	Introduction	43
5.2	Related Work	44
5.3	Angle I: Types of data	46
5.3.1	Data Collection	46
5.3.2	Statistics of the Dataset	48
5.3.3	Comparison with Existing Datasets	49
5.4	Angle II: Benchmark	49
5.4.1	Problem Formulation	49
5.4.2	Existing Methods	50
5.4.3	Proposed Method	50
5.5	Angle III: Tasks and Results	53
5.5.1	Types of tasks	53
5.5.2	Evaluation of Traditional Tasks	53
5.5.3	Results and Discussion	54
5.5.4	Thumbnail Generation	56
5.6	Conclusion	57
II	Robustness of Multimodal Models under Perturbations	58
6	Robustness of Multimodal Models under Distribution Shift	59
6.1	Introduction	59
6.2	Related Work	60
6.3	Multimodal Robustness Benchmark	61
6.3.1	Image Perturbation	62
6.3.2	Text Perturbation	63
6.4	Experiments	64
6.4.1	Evaluation Tasks, Datasets and Models	65
6.4.2	Evaluation Metrics	65
6.4.3	Robustness Evaluation under Distribution Shift	66
6.5	Discussion	72
6.6	Conclusion	73
III	Generalization to Interactive Multimodal Environment	75
7	Language-based Scene Summarization for Embodied Policy Learning	76
7.1	Introduction	76
7.2	Related Work	77
7.3	Proposed Method	78
7.3.1	Problem Formulation	79
7.3.2	APM: Decoding Language Information into Executable Action Plans	80
7.3.3	Training Pipeline	80
7.3.4	Fine-tuning APM with IL and RL	81

7.4	Experiments	82
7.4.1	Environments and Metrics	82
7.4.2	Datasets	83
7.4.3	Experimental Setup	85
7.5	Results and Discussions	86
7.5.1	Model Performance with IL Fine-tuning	86
7.5.2	Model Performance with RL Fine-tuning	87
7.5.3	Ablation Study	88
7.6	Limitations and future directions	91
7.7	Conclusion	91
8	Entity-Centric VQA by Retrieval Augmented Multimodal LLM	92
8.1	Introduction	92
8.2	Related Works	94
8.3	SnapNTell Dataset	95
8.3.1	Entity Categorization	95
8.3.2	Image collection	95
8.3.3	Knowledge-intensive Question-Answer Pairs	96
8.3.4	Statistics and Analysis of Our Dataset	98
8.3.5	Comparison with Existing VQA Datasets	99
8.4	Proposed Method	101
8.4.1	Retrieval Augmentation	101
8.4.2	Entity-centric Knowledge-based Answer Generation	103
8.5	Experiments and Results	103
8.5.1	Experimental Setup	103
8.5.2	Results and Discussion	104
8.5.3	Ablation Study	105
8.5.4	Human Evaluation Results	106
8.5.5	Discussions	107
8.6	Conclusion	108
IV	Cross-domain Applications in Healthcare	109
9	Detect Cardiovascular Disease Through Language Models	110
9.1	Introduction	110
9.2	Related Work	111
9.3	Proposed Method	112
9.4	Dataset and Preprocessing	113
9.5	Experiments	116
9.5.1	Experimental Settings	116
9.5.2	Results	117
9.5.3	Ablation Study	117
9.5.4	Limitations	119
9.6	Conclusion	120

10	Inner Alignment between Brain Signals and Human Languages	121
10.1	Introduction	121
10.2	Related Work	123
10.3	Proposed Method	124
10.3.1	Overview of Model Architecture	124
10.3.2	Hierarchical Transformer Encoders	125
10.3.3	Cross Alignment Module	125
10.4	Experiments	126
10.4.1	Downstream Tasks	126
10.4.2	Datasets	127
10.4.3	Experimental Settings	127
10.4.4	Experimental Results and Discussions	128
10.4.5	Ablation Study	128
10.4.6	Analysis	129
10.5	Conclusion	133
11	Clinical Retrieval System for Cardiovascular Magnetic Resonance Imaging	134
11.1	Introduction	134
11.2	Related Work	136
11.3	Proposed Method	137
11.3.1	Model Architecture	137
11.3.2	Learning Objectives	139
11.4	Our CMR Dataset	140
11.4.1	Specific Characteristics of the CMR Data	141
11.4.2	Data Preprocessing and Filtering	141
11.4.3	Statistics of the Data	142
11.4.4	Comparison with Existing CMR Datasets	142
11.5	Experiments	143
11.5.1	Experimental Setting	143
11.5.2	Experimental Results on the Retrieval Task	144
11.5.3	Experimental Results on Image Classification Task on our Cardiomy-	
	opathies Dataset	145
11.5.4	Experimental Results on Image Classification Task on ACDC Dataset	147
11.6	Discussion	147
11.7	Conclusions	149
V	Conclusions and Future Directions	150
12	Conclusions	151
12.1	Summary of Contributions	151
12.2	Broader Impact	153
12.3	Future Directions	154
12.3.1	Follow-up Work	154
12.3.2	Broader Discussion	155

List of Figures

1.1	Multimedia summarization: providing summaries can significantly lead to better search engines and recommendation systems.	2
1.2	Content creation: text-to-image models are easy to attack and generate incorrect results. For example, they might generate an image of a dog that is not white or a scene without grass or trees, despite these elements being specified in the input description but with a few common perturbations. ("Keyboard" simulates the mistakes made while using a keyboard.)	3
1.3	Household Robot: the market is increasing with few practical solutions [3].	4
1.4	Healthcare applications: the healthcare treatment cost is a financial burden but still increasing.	4
1.5	Healthcare applications: different types of healthcare data, from [372].	5
1.6	Thesis contributions to multimodal alignment, robustness, and generalizability.	5
3.1	Comparison with previous work: We proposed a segment-level cross-domain alignment model to preserve the structural semantics consistency within two domains for the MSMO task. We solve an optimal transport problem to optimize the cross-domain distance, which in turn finds the optimal match.	17
3.2	SCCS at work: A real example of the summarization process given by our SCCS method. Here we conduct OT-based cross-domain alignment to each keyframe-sentence pair, and a smaller OT distance means better alignment. (For example, the best-aligned text and image summary (0.08) delivers the flooding content clearly and comprehensively.)	18
3.3	SCCS framework: (a) The computational framework of the SCCS model, which takes multimodal inputs (videos & text documents) and generates multimodal summaries. The framework includes five modules: video temporal segmentation, visual summarization, textual segmentation, textual summarization, and multimodal alignment. (b) The structure of the video segmentation encoder. (c) The architecture of the textual segmentation module. (d) The multimodal alignment module for multimodal summaries.	19
3.4	OT coupling: The OT coupling shows sparse patterns and specific temporal structure for the embedding vectors of ground-truth-matched video and text segments.	26
4.1	Comparison of temporal-pairwise cosine distance on visual features: (TOP) a Livestream video, (BOTTOM) a TVSum video (Blue & Green: distance; Red: segment boundaries).	29
4.2	Example of Livestream videos.	31

4.3	Histogram of MultiLive video length distribution and transcript length distribution (y-axis: number of videos).	32
4.4	(a) Visual features of a Livestream video; (b) Visual features of a TVSum video, where different colors represent different segments within one video.	33
4.5	Dendrogram result of one Livestream video by hierarchical clustering of visual features, where the numbers below the bottom layer represent the number of images belonging to the corresponding sub-tree.	34
4.6	LiveSeg framework: The framework of LiveSeg for unsupervised multimodal Livestream video temporal segmentation.	35
4.7	Graphical model of HDP-HSMM	38
4.8	Comparison of boundary candidates by different methods, from top to bottom: (1) HCA, (2) TransNet V2, (3) Hecate, (4) OSG, (5) LGSS, (6) ours (LiveSeg), and (7) Human Annotations.	40
4.9	Segmentation performance with different parameters (Red: precision; Blue: recall; Green: F1-score).	41
5.1	Task comparison: (a) traditional video summarization, (b) text summarization, and (c) MSMO tasks.	44
5.2	MMSum: The design of the proposed MMSum dataset is driven by research and application needs.	46
5.3	MMSum categories: The 17 main categories of the MMSum dataset, where each main category contains 10 subcategories, resulting in 170 subcategories in total.	47
5.4	MMSum statistics: The statistics of the MMSum dataset, which show the distribution of (a) video duration; (b) number of segments per video; (c) segment duration; (d) number of words per sentence.	47
5.5	Our model comprises two modules: the segmentation module and the summarization module.	50
5.6	Comparison of GT thumbnails and our generated ones.	56
6.1	Multimodal models are sensitive to image/text perturbations (original image-text pairs are shown in blue boxes, perturbed ones are in red). Image captioning (Top): Adding image perturbations can result in incorrect captions, e.g., the tabby kitten is mistakenly described as a woman/dog. Text-to-image generation (bottom): Applying text perturbations can result in the generated images containing incomplete visual information, e.g., the tree is missing in the two examples above.	60
6.2	Examples of our 17 image perturbations. The original image is taken from the COCO dataset and shown on the top left.	63
6.3	OT alignment visualization between text and perturbed images , where <i>pixelate</i> and <i>zoom blur</i> are two high-effective image perturbation methods, <i>brightness</i> and <i>glass blur</i> are two low-effective ones.	68
6.4	OT alignment visualization between perturbed text and images, where <i>keyboard</i> and <i>character replace</i> are two high-effective text perturbation methods, <i>insert punctuation</i> and <i>formal</i> are two soft ones.	68

6.5	(a) Image captioning results of BLIP; (b) Image captioning results of GRIT; (c) Grad-CAM visualizations on the cross-attention maps corresponding to individual words under image perturbations, where <i>zoom blur</i> and <i>pixelate</i> perturbed images show worse word-image attention alignment than the <i>brightness</i> perturbed image. For example, in <i>zoom blur</i> and <i>pixelate</i> , the “door” and “glasses” words’ attention maps are not matched with the correct image patches, while in <i>pixelate</i> , all words’ attention maps match correctly.	69
6.6	(a) Text-to-image generation results of Stable-diffusion in terms of (a) FID scores; (b) CLIP-FID scores. Since both scores are the lower, the better; a higher bar indicates the model is less robust to a particular perturbation. (c) Grad-CAM visualizations on the cross-attention maps corresponding to perturbed captions and images generated by perturbed captions. We use the original unperturbed word query to visualize the attention map. In <i>keyboard</i> , the hydrant is missing; in <i>word deletion</i> , the color of the hydrant is incorrect, but no object is missing; in <i>casual</i> , the attention map perfectly matches the generated images, which shows character-level perturbations could be more effective than word level and sentence-level perturbations.	71
6.7	Left: Missing Object Rate (MOR) metric calculation. Right: Comparison of detection results between GT-caption-generated images (top) and perturbed-caption-generated images (bottom).	71
7.1	The overall architecture of our approach, which includes a scene understanding module (SUM) and an action prediction module (APM). The agent takes pure visual observations and encodes the information as latent language, then the language is transferred to APM for action generation. APM fine-tuned on VirtualHome can generate executable action plans directly.	78
7.2	Top-down views of 7 different environments from VirtualHome.	83
7.3	‘AUTO’, ‘FIRST PERSON’, ‘FRONT PERSON’ views.	85
7.4	Comparison with baselines in the <u>imitation learning</u> setting evaluated by the execution rate.	86
8.1	Comparing SnapNTell with existing methods reveals a distinctive focus. In the SnapNTell benchmark, the answers are predominantly entity-centric , characterized by a greater depth of knowledgeable information pertaining to the specific entity depicted in the image as the answer.	93
8.2	Statistics of number of entities in each category.	95
8.3	Collecting images for building the evaluation dataset. Licenses: CC Publicdomain, CC Attribute, AA Sharealike, CC Noncommercial, or CC Nonderived licenses. Metadata: image URLs, source page URLs, renamed image names, and the corresponding Wikipedia page URL.	96
8.4	The information collected during dataset building, i.e., from Wikipedia for each entity, which includes the summary of the general introduction, toponym, location information, and so on.	98
8.5	Average pageview per entity within each category, where average pageview is defined as the sum of pageviews/ number of entities.	99
8.6	Statistics of all pageviews for all categories.	99

8.7	Comparison with existing VQA datasets, where previous VQA datasets mostly focus on freeform answers (such as yes/no for verification questions and choice for selection questions).	101
8.8	Examples from our SnapNTell dataset.	102
8.9	SnapNTell model: Our SnapNTell model architecture takes an image-question pair as input. It begins with retrieval augmentation to source relevant information about the entity in the image. This information, along with the question, feeds into the word embedding layer. Text embeddings merge with image-projected embeddings before entering the LLM, culminating in a knowledgeable answer as the output.	102
8.10	Human evaluation results on pairwise comparisons (% win, tie, lose) with baseline outputs <i>against</i> the manually annotated ground-truth from SnapNTell.	107
9.1	Example of 12-lead ECG [444].	111
9.2	The architecture of our model. The Transformer encoder takes input ECG to generate ECG features as the input to LLM, where LLM transforms it into generated embeddings. An Optimal Transport (OT)-based loss objective is formulated on generated embeddings and ground-truth embeddings for the model update.	113
9.3	ECG data in the format of time series and spectrum.	114
9.4	Filtered ECG data in the format of time series and spectrum.	114
9.5	Detecting R peaks in the ECG signals.	115
9.6	Extracted ECG beats divided by R peaks.	115
10.1	Commonly used techniques for recording brain activity [1].	122
10.2	Three paradigms of EEG and language alignment.	122
10.3	The architecture of our model, where EEG and language features are coordinately explored by two encoders. The EEG encoder and language encoder are shown on the left and right, respectively. The cross-alignment module is used to explore the connectivity and relationship within two domains, while the transformed features are used for downstream tasks.	123
10.4	TSNE projection comparison of untransformed & transformed features of ZuCo dataset, where different colors represent different classes.	130
10.5	Negative word-level alignment.	130
10.6	Positive word-level alignment.	130
10.7	Negative and Positive sentence-level alignment of ZuCo dataset.	130
10.8	Positive and Negative word brain topologies (Sentiment Analysis)	132
11.1	Examples of Cardiac magnetic resonance (CMR) images. CMR comprises hundreds of images of differing views and image types, the large majority of which do not contain an indication of pathology. The corresponding report is an analysis of all these images, resulting in difficulty in aligning images with text reports.	135
11.2	The overall architecture of our model, where the visual encoder processes sequences of CMR images and the text encoder processes the text from the "impression" section of the corresponding reports.	138
11.3	Data preprocessing pipeline of our CMR dataset.	141

11.4	Example of CMR image sequences constructed by $\text{CINE}_{\text{lax-sax}} + \text{LGE}_{\text{lax-sax-2ch-3ch}}$, where (\cdot) represents the number of images of each type-view combination. For $\text{CINE}_{\text{lax-sax}}$, each frame represents the time dimension. For LGE_{sax} , each frame corresponds to the depth dimension, and for $\text{LGE}_{\text{lax-2ch-3ch}}$, each image is duplicated to be consistent with LGE_{sax}	144
11.5	t-SNE visualization of zero-shot visual embeddings on the ACDC dataset.	148
11.6	t-SNE visualization of learned visual embedding by CMRformer on ACDC.	148

List of Tables

1.1	Thesis topics.	9
3.1	Comparison with multimodal baselines on the VMSMO dataset. The absolute performance comparison with the baseline MSMO method is marked in red (better) and blue (worse).	24
3.2	Comparisons on the Daily Mail and CNN datasets. The absolute performance comparison with the baseline MSMO method is marked in red (better) and blue (worse).	25
3.3	Human evaluation results.	25
3.4	Factual consistency evaluation results.	26
4.1	Distribution of video duration.	31
4.2	Distribution of transcript length.	31
4.3	Comparison of MultiLive with existing datasets. SLR: scene length ratio.	32
4.4	Comparison of different types of videos.	33
4.5	Example of livestream transcripts.	34
4.6	Comparison of segmentation results.	40
4.7	Comparison of performance with different interval ω_t	41
4.8	Ablation study of different components.	42
4.9	Comparison with SOTA unsupervised baseline on traditional video summarization datasets.	42
5.1	Comparison of the modality of different <u>summarization tasks and datasets</u> . Difference between traditional multimodal summarization and MSMO: traditional multimodal summarization still outputs a single-modality summary, while MSMO outputs both modalities' summaries. Public Availability means whether the data is still publicly available and valid. Structural Summaries means available summaries of each segment, not just for the whole video.	48
5.2	Comparison with existing video summarization and multimodal summarization datasets.	48
5.3	Comparison of video temporal segmentation results.	54
5.4	Comparison of video summarization results (whole-video setting and segment-level setting).	54
5.5	Comparison of textual summarization results (whole-video setting and segment-level setting).	55
5.6	Comparison of MSMO results.	55

6.1	Image perturbations.	62
6.2	Text perturbations.	64
6.3	Example of our 16 text perturbations. The original text is taken from the COCO dataset and denoted as clean in the first row.	65
6.4	Evaluation tasks, datasets, models and metrics used in our study.	66
6.5	Image-text retrieval. [Top] Robustness evaluations on Flickr30k-IP and COCO-IP. [Bottom] Robustness evaluations on Flickr30k-TP and COCO-TP datasets. We report averaged RSUM where the most effective perturbation results are marked in bold, and the least effective perturbation results are underlined. The MMI impact score is marked in blue; the lower, the better.	67
6.6	Visual reasoning (VR) and visual entailment (VE): [Top] Robustness evaluations for NLVR2-IP and SNLI-VE-IP datasets. [Bottom] Robustness evaluations for NLVR2-TP and SNLI-VE-TP. We report the averaged accuracy where the most effective perturbation results are marked in bold, and the least effective ones are underlined. The MMI impact score is marked in blue, the lower the better.	70
6.7	Missing Object Rate (MOR) results of Stable Diffusion. The most effective perturbation results are marked in bold, and the least effective ones are underlined. The results show that more objects are missing from the images generated by character-level perturbed captions.	72
6.8	Top1 classification accuracy of unimodal vision models. The most effective perturbation results are marked in bold and the least effective ones are underlined.	73
7.1	Results by different SUM fine-tuned by <u>imitation learning (IL)</u> objective, where BERT serves as APM. The results are shown on 7 different environments in VirtualHome and also the average performance. The best result in each environment and each SUM model is marked in black and bold. The best SUM result with the highest average performance across 7 environments is marked in orange and bold.	87
7.2	Execution Rates by different SUM fine-tuned by <u>REINFORCE</u> , where BERT serves as APM. The results are shown on 7 different environments and also the average performance. The best results are marked in bold.	87
7.3	Results by different APM fine-tuned by <u>imitation learning (IL)</u> loss objective. The results are shown by the average of 7 different environments in VirtualHome. The best results are marked in bold.	88
7.4	Results by different APM fine-tuned by <u>REINFORCE</u> loss objective, averaging on 7 different environments. The best results are marked in bold.	88
7.5	Comparison of episode success rate.	89
7.6	Our fine-tuning results for different SUM/APM configurations in in-distribution and novel tasks, as well as using REINFORCE and imitation learning strategies. We measure the performance based on the episode success rate.	90
7.7	Comparison action execution rates in zero-shot and fine-tuned settings using both REINFORCE and Imitation Learning.	90
7.8	Comparison episode success rate in zero-shot and fine-tuned settings using both REINFORCE and Imitation Learning.	91

8.1	Filtering statistics of the entity dataset. [1st Wiki filtering]: removing ones without a wiki page. [2nd Google filtering]: removing ones without enough images via google search API. [3rd Wiki filtering]: removing entity names with ambiguous wiki pages.	97
8.2	Types of questions.	97
8.3	Comparison with existing VQA datasets <i>Knowledge</i> means the QA pairs are knowledgeable, not simple yes/no answers or selection questions. <i>Entities</i> means whether there are fine-grained entities specifically contained in answers. <i>Categorization</i> means the entities are categorized, not randomly crawled online.	100
8.4	More detailed comparison with existing knowledge-based VQA datasets. <i>Anonymity</i> means whether the question already contains a knowledge clue related to the entity in question. (* Unclear)	100
8.5	Performance comparison of different approaches on the SnapNTell dataset.	104
8.6	Effectiveness of evaluation metrics.	105
8.7	Ablation study on the effectiveness of entity detection (ED).	105
8.8	Ablation study on head/torso/tail entities, where RA is short for Retrieval Augmentation and Δ is the performance difference of with and without RA.	106
8.9	Ablation on the <u>accuracy</u> performance of different VQA datasets (A lower result means the task is <u>more</u> complicated to solve).	106
9.1	ECG statistical features in the frequency domain.	115
9.2	Statistics of the processed ECG data.	116
9.3	Comparisons of different backbones on Text generation (TG) and Disease detection (DD). (BERT as LLM)	116
9.4	Comparisons with supervised baselines (DD).	116
9.5	Examples of comparison on generated reports (marked as Predicted-X) and ground-truth reports (marked as GT-X).	117
9.6	Comparisons of different LLMs on Text generation (TG) and Disease detection (DD). (Transformer as the encoder).	117
9.7	Comparisons with different backbones on the text generation task, where BERT is used as LLM.	118
9.8	Comparisons with different backbones on the disease detection task, where BERT is used as LLM.	118
9.9	Comparisons of different number of transformer layers on the text generation task, where BERT is used as LLM.	118
9.10	Comparisons of different numbers of transformer layers on the disease detection task, where BERT is used as LLM.	118
9.11	Ablation study of different transformer layers.	119
10.1	Comparison with baselines on K-EmoCon dataset for Sentiment Analysis.	128
10.2	Comparison with baselines on Zuco dataset for Sentiment Analysis (SA) and Relation Detection (SD).	129
10.3	Ablation results on the components in the CAM module (best results in bold).	131
10.4	Comparison of performance on K-EmoCon dataset with different physiological signals as inputs on the Sentiment Analysis task.	131

11.1	Statistics of length of impression sections from text reports.	142
11.2	Statistics of the number of images of each study.	142
11.3	Comparison with existing CMR datasets.	143
11.4	Model parameters in the experiments.	144
11.5	Experimental results for retrieval experiments. (·) represents the number of input frames. Zero-Shot evaluation was done using $CINE_{lax-sax} + LGE_{lax-sax-2ch-3ch}$	145
11.6	Linear probing results on the Cardiomyopathies dataset for the downstream disease classification task.	146
11.7	Comparison of the image classification results on the ACDC dataset, where AUC is computed in a one-vs-rest manner and both F1 score and AUC are micro-averaged.	147
12.1	Work in topics	153

List of Algorithms

1	Compute Alignment Distance	22
2	Fine-tuning SUM	80
3	Fine-tuning APM with Imitation Learning	80
4	Fine-tuning APM with REINFORCE	81

List of Acronyms

MSMO Multimodal summarization with multimodal output	16
OT Optimal Transport	xvi
SCCS Semantics-Consistent Cross-domain Summarization	17
CDA cross-domain alignment	21
WD Wasserstein distance	12
MAP mean average precision	23
LM language model	24
GWD Gromov Wasserstein Distance	35
CCA Canonical Correlation Analysis	13
HDP-HSMM Hierarchical Dirichlet Process Hidden semi-Markov Model	37
SBP stick-breaking process	37
RMSE Root Mean Squared Error	54
SSIM Structural Similarity Index	54
SRE Signal reconstruction error ratio	54
SAM Spectral angle mapper	54
OOD Out-of-Distribution	61
ID In-Distribution	66
VR visual reasoning	65
VE visual entailment	65
ZS zero-shot	66
LLMs Large Language models	76
NLP Natural Language Processing	77
IL imitation learning	76

RL Reinforcement Learning	78
CE cross-entropy	81
MLM masked language model	81
std standard deviation	89
VQA Visual Question Answering	8
KG Knowledge Graph	103
ECG Electrocardiography	110
FFT Fast Fourier transform	114
EEG Electroencephalography	121
CMR cardiac magnetic resonance	134

Chapter 1

Introduction

The world is fulfilled with various signals such as images, videos, audio, text, sensor signals, and so on. Humans naturally perceive the world through multiple senses, such as sight, sound, touch, and more. This multi-modal approach is a fundamental aspect of human cognition and has become a cornerstone in the development of artificial intelligence (AI). In the current era, where technology is increasingly driven by vast amounts of data, the concept of multimodal intelligence has gained prominence. This paradigm combines various forms of data, such as language, visual information, physiological signals, and others, to create more comprehensive and nuanced AI systems.

Despite the wealth of data available, there are currently no comprehensive tools for analyzing this data and leveraging the patterns within it. Identifying patterns across different modalities is essential for utilizing them to address real-world, domain-specific challenges. For motivation examples: (1) Summarization and Recommendation: The automatic generation of summaries for multimedia news or providing introductions to online videos can significantly enhance the performance of search engine and recommendation systems for online content. For instance, more than 500 hours of video are uploaded to YouTube every minute, most without summaries. The absence of accurate summaries makes it challenging to develop search engines and recommendation systems that efficiently help users find the content they desire. (2) Content Creation: Text-to-image generation models often produce incorrect, unclear, or biased content. Developing more robust models for real-world applications is an urgent issue. (3) Household Robots: The market for household robots, aimed at assisting people with disabilities in daily tasks, was valued at USD 10.3 billion in 2023. These robots aim to offer substantial and efficient support for routine tasks through various interaction methods, including visual perception, verbal instructions, and speech dialogue, making the utilization of multimodal information to enhance the robot's performance and reliability a complex challenge. (4) Healthcare Applications: For example, the diagnosis of chest pain in emergency departments (ED) alone currently incurs an estimated cost of 10 to 12 billion per year in the US. Developing a solution that provides cost-efficient patient care using multimodal healthcare data could be highly beneficial for society.

This thesis is motivated by these applications. The problems studied in this thesis are abstracted from the common challenges across these applications. In the following, we will first present a few motivating application. We will describe the problems and general approaches to the challenges.

1.1 Motivation and Challenges

Multimodal Learning has advanced quickly in recent years with numerous applications in different fields, i.e., multimedia image/video and language understanding [99, 179, 389, 580], embodied learning [38, 178, 192, 294], healthcare and psychology [146, 267], and many more.

Multimodal learning is essential in real-world applications as it empowers systems to process and comprehend information from a variety of sources. In these scenarios, it is very important to understand the patterns in the data such as alignment, robustness, and generalizability. Our goal is to develop algorithms and build datasets for learning from multiple modalities, and we list here a few motivating applications.

Multimedia search engine and recommendation systems The digital world is overflowing with multimedia content, such as videos. For instance, with more than 500 hours of video uploaded to YouTube every minute, many without detailed annotations, creating multimedia search engines and recommendation systems is a challenging task. Traditional search engines rely on titles, text descriptions, or video tags, but for more precise search results, additional content like summaries is needed. However, the number of videos with summaries is small compared to the volume of new uploads, making it crucial to find ways to generate summaries for videos to enhance search engine performance and better recommendations for the users.

One promising solution is Multimodal Summarization with Multimodal Output (MSMO), which has gained traction in recent years [52, 172, 210, 309, 576]. MSMO aims to automatically generate keyframes and key textual summaries for media news or online videos. These applications can deliver concise summaries of multimedia content, which can significantly improve the development of search engines and recommendation systems, helping users find the content they are looking for more effectively.

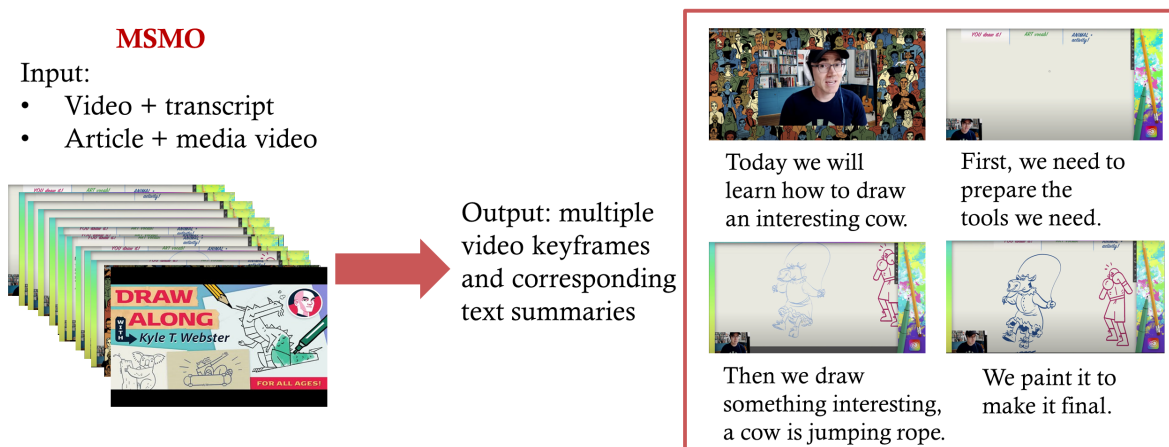


Figure 1.1: Multimedia summarization: providing summaries can significantly lead to better search engines and recommendation systems.

Figure 1.1 illustrates a potential solution for addressing the randomness of videos on the internet, which can aid in the development of improved search engines and recommendation systems. In this context, we are particularly interested in addressing the following critical issues:

- How do we explore the inner semantic alignment between different domains?

- How can the learned alignment help advance multimodal applications, such as providing better summarization results?

Content Creation As the field of text-to-image generation models [317, 389] evolves, these technologies are increasingly being utilized to assist in the creation of creative content. They serve a wide range of users, from professional designers to individuals without specific domain expertise, effectively enhancing the efficiency and creativity of content production. Despite their growing popularity, these models encounter several challenges. One of the main issues is their vulnerability to attacks, which can lead to the generation of incorrect, unclear, or biased content. This highlights the pressing need for the development of more robust models that can withstand such vulnerabilities and perform reliably in real-world applications. Addressing this concern is essential for ensuring the utility of text-to-image generation technologies in various creative fields.

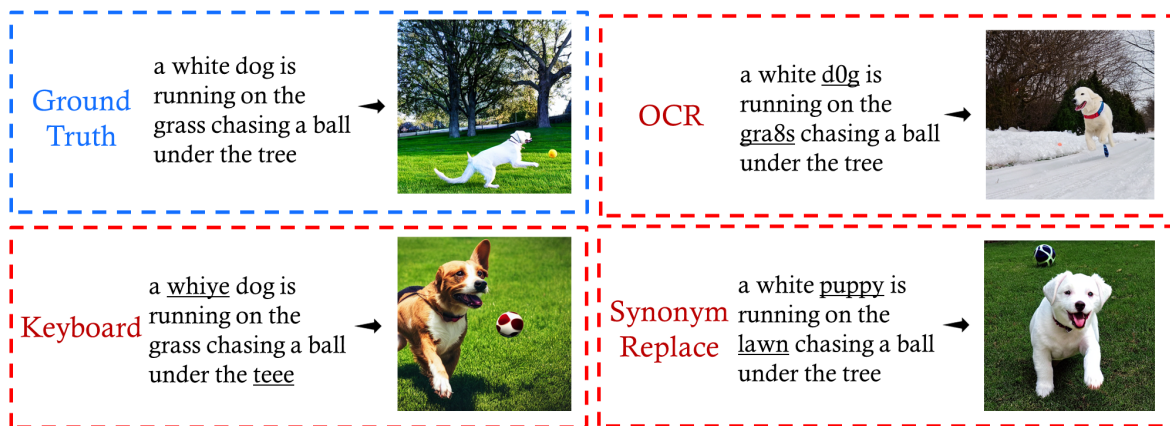


Figure 1.2: Content creation: text-to-image models are easy to attack and generate incorrect results. For example, they might generate an image of a dog that is not white or a scene without grass or trees, despite these elements being specified in the input description but with a few common perturbations. (“Keyboard” simulates the mistakes made while using a keyboard.)

Figure 1.2 illustrates a text-to-image generation example, where applying keyboard typos, OCR errors, or synonym replacements to the original sentence, can lead to generated images containing incomplete visual information. In this context, we are interested in exploring:

- How robust are the multimodal models in the presence of noises??
- How can we improve the models’ robustness in real-world applications?

Household Robot In 2023, the industry focusing on household robots designed to aid individuals with disabilities in their daily activities was estimated to be worth USD 10.3 billion. These robots aim to offer substantial and efficient support for routine tasks through various interaction methods, including visual perception, verbal instructions, and speech dialogue, making the utilization of multimodal information to enhance the robot’s performance and reliability a complex challenge.

Large Language Models (LLMs) have recently made significant advancements in supporting robotic learning for intricate domestic tasks such as complex household management. Nonetheless, the efficacy of these pre-trained LLMs depends greatly on templated text data tailored to specific domains, a requirement that may not be practical for real-world robotic learning scenarios that

involve image-based observations. Furthermore, current LLMs that process textual information are not designed to adapt through non-expert interactions with environments. Consequently, the pressing question arises: How can we leverage multimodal data to develop better household robots that can more effectively assist humans in the complex environment?

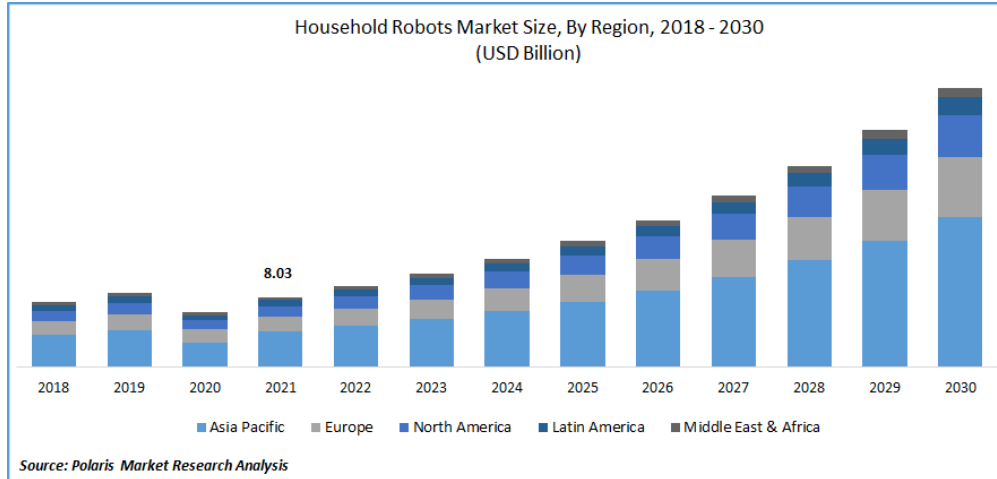


Figure 1.3: Household Robot: the market is increasing with few practical solutions [3].

Healthcare applications Providing high-quality and efficient patient treatment is a longstanding problem. For example, in the current practice of cardiovascular disease, patients presenting with chest pain to the emergency department (ED) constitute a diagnostic and logistic challenge as chest pain can be caused by an extensive variety of disorders [14]. Diagnostic tests and decision algorithms play a critical role in speeding up the appropriate triage of chest pain patients in the ED, and preventing unnecessary hospitalization of patients with non-critical disorders. In current practice, about half of the patients presenting with chest pain can be discharged from the ED, and only 5.5 % of all ED visits lead to serious diagnosis [175]. However, the diagnosis of chest pain in the ED now incurs an estimated cost of 10 to 12 billion per year in the U.S., representing a significant financial burden for both patients and society. The pressing issue, therefore, is how to provide cost-efficient patient care using multimodal healthcare data.

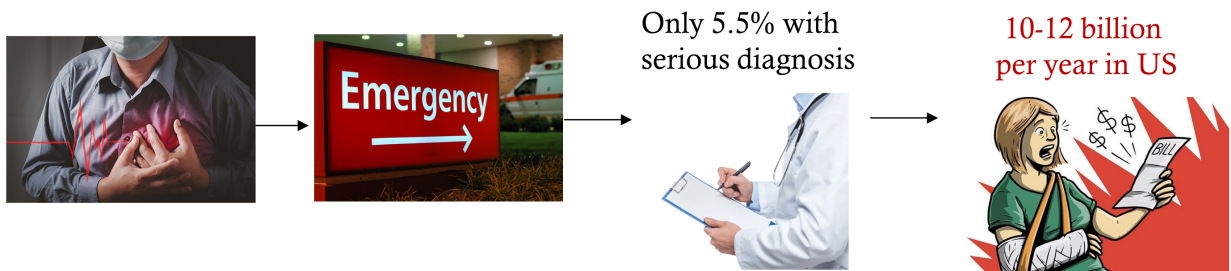


Figure 1.4: Healthcare applications: the healthcare treatment cost is a financial burden but still increasing.

Since the data available in the robotics and healthcare domains are less abundant compared to the vast amounts of multimedia image/video or text data, there are inherent challenges, even though healthcare data, as illustrated in Figure ??, comes in various types. However, the volume of

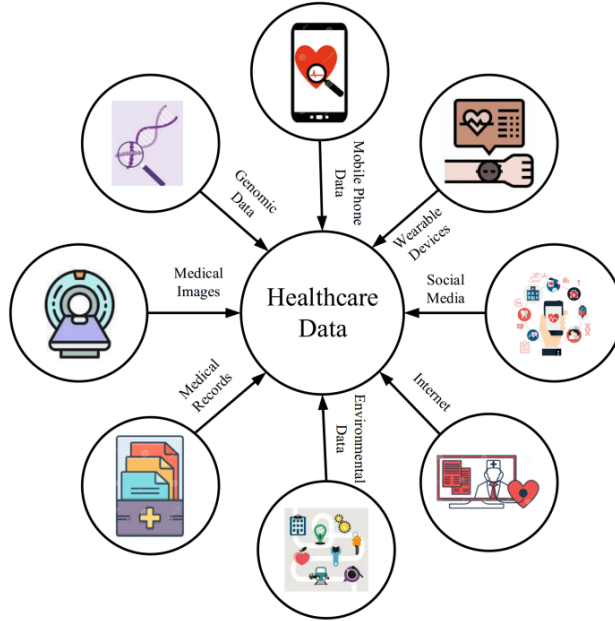


Figure 1.5: Healthcare applications: different types of healthcare data, from [372].

data within each type is still significantly lower than that of image-text data. This leads to typical problems including:

- How do we generalize the knowledge of one learned domain to another unlearned domain?
- How to generalize from data-rich domain to data-scarce domain?

1.2 Thesis Overview

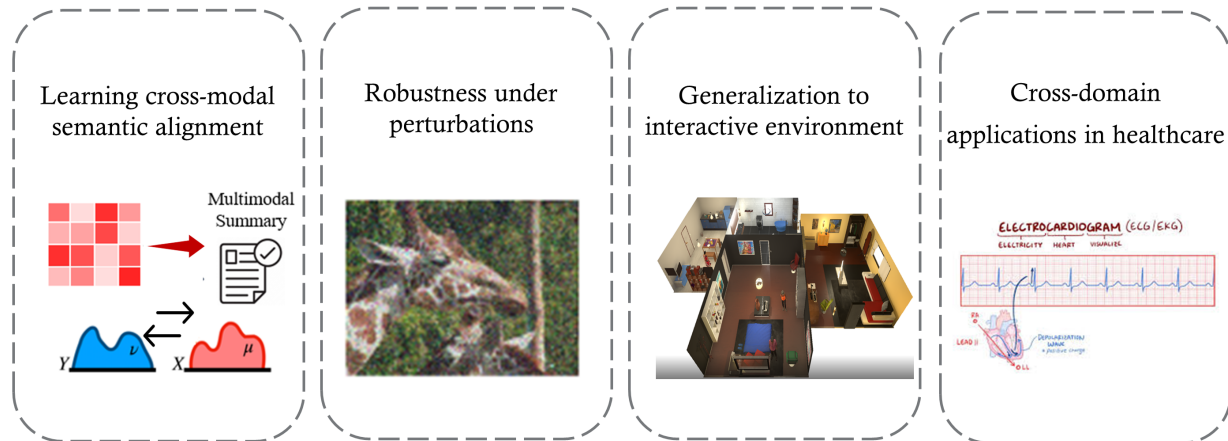


Figure 1.6: Thesis contributions to multimodal alignment, robustness, and generalizability.

Given this context, several critical questions need to be addressed to improve the alignment, robustness, and generalizability of multimodal learning:

1. **Exploring Inner Semantic Alignment** How do we explore the inner semantic alignment between different domains? How can the learned alignment help advance multimodal ap-

plications? Understanding the relationships between different modalities and how they can complement each other is crucial for developing more effective multimodal systems.

2. **Robustness of Multimodal Models** How robust are the multimodal models? How can we improve the models' robustness in real-world applications? Ensuring that multimodal models can handle diverse and potentially noisy inputs is essential for their reliability and effectiveness in practical applications.
3. **Generalization Across Domains** How do we generalize the knowledge of one learned domain to another unlearned domain? Developing methods that allow models to transfer learned knowledge to new, unseen domains is key to creating more versatile and adaptable multimodal systems.

In this thesis, we aim to answer each of these three questions, aiming to enhance the performance, robustness, and interpretability of multimodal models in real-world scenarios, ultimately contributing to the advancement of multimodal intelligence.

Chapter 2: multimodal semantic alignment This chapter focuses on achieving effective multimodal semantic alignment, facilitating seamless connections between language and visual modalities. To accomplish this objective, the following sub-objectives are pursued:

- **Multimodal alignment:** We explore establishing rich connections between language and image/video data. By aligning the semantic content of language with visual elements, the resulting models can possess a more nuanced understanding of the underlying concepts. In [361], we propose a Semantics-Consistent Cross-domain Summarization (SCCS) model, which leverages optimal transport alignment combined with visual and textual segmentation to achieve multimodal summarization. In [347], we explore the alignment between visual and language domains specifically for the task of temporally segmenting long Livestream videos, which can establish the basis for Livestream video understanding tasks and can be extended to many real-world applications.
- **Interpretability:** We delve into the application of Optimal Transport-based approaches to learn cross-domain alignment, enabling models to provide interpretable explanations of their multimodal reasoning process [361, 363]. The optimal transport coupling can reveal the underlying similarity and structure, which further helps to explain the correspondence between the text and image data.
- **New datasets:** We propose a new dataset, MMsum [357], to solve the problems within existing datasets, such as insufficient maintenance, data inaccessibility, limited size, etc. MMsum is specifically designed to cater to a wide range of tasks, with a particular emphasis on MSMO, with diverse categorization.

Chapter 3: multimodal robustness In Chapter 3, we aim to address the challenge of multimodal robustness, particularly under perturbations. As real-world scenarios often involve variations in data distributions, it is crucial to ensure that multimodal models can maintain their performance across diverse environments. To tackle this challenge, the research will focus on the following areas:

- **Robustness evaluation:** In [363], we build a comprehensive evaluation benchmark specifically designed to assess the robustness of multimodal models. By simulating various distribution shifts and measuring the model's performance across different scenarios, we aim to gain a

deeper insight into the model’s adaptability and vulnerabilities. Understanding how multimodal models perform under diverse conditions is crucial for developing more reliable and robust systems that can effectively handle the complexities of real-world data.

Chapter 4: multimodal generalizability capabilities in interactive environments In Chapter 4, we aim to explore the multimodal generalizability capabilities in interactive environments, particularly in the domains of language grounding. The following sub-objectives are pursued:

- Language grounding in robot learning: In [350], we introduce a novel learning paradigm that generates robots’ executable actions in the form of text, derived solely from visual observations. Our proposed paradigm stands apart from previous works, which utilized either language instructions or a combination of language and visual data as inputs.
- Retrieval-augmented generation (RAG): In [353], we work on a novel task for entity-centric VQA to assess the proficiency of models in accurately identifying and generating responses that exhibit a deep comprehension of these identified entities. We propose a retrieval-augmented multimodal LLM, devised as a baseline model capable of undertaking the SnapN-Tell [353] task, which is scalable, effective, and explainable.

Chapter 5: cross-modal applications in healthcare Finally, in Chapter 5, we explore whether the learned knowledge can be transferred to the clinical domain. We aim to explore the cross-modal applications in healthcare.

- ECG-to-text generation: In [349], we aim to bridge the gap by transferring the knowledge of LLMs to clinical Electrocardiography (ECG) for textual diagnosis report generation and zero-shot disease detection. Our approach is able to generate high-quality cardiac diagnosis reports and also achieves competitive zero-shot classification performance even compared with supervised baselines, which proves the feasibility of transferring knowledge from LLMs to the cardiac domain.
- Connectivity between human language and brain signals: In [146], we explore the relationship and dependency between EEG and human language to reveal the inner connection. Our findings on word-level and sentence-level EEG-language alignment show the influence of different language semantics as well as EEG frequency features.
- Clinical retrieval system: In [351], we design a retrieval system that can automatically match the input Cardiovascular Magnetic Resonance (CMR) Imaging to the most similar records in the database. This functionality can significantly aid in diagnosing diseases and reduce physicians’ workload, which can provide patients with better treatment.

1.3 Summary of Contributions

Multimodal intelligence, where AI systems exhibit intelligent behaviors by leveraging data from multiple modalities (text, visual, audio, etc.), has emerged as a key concept in today’s data-driven era. This cross-modal approach finds diverse applications and transformative potential across industries. By fusing heterogeneous data streams, multimodal AI generates representations more akin to human-like intelligence than traditional unimodal techniques. In this thesis, we aim to propel the field of multimodal AI forward by enhancing alignment, robustness, and generalizability, thus paving the way for more sophisticated and efficient multimodal AI systems.

Algorithms Our work has focused on:

- Multimodal alignment [347, 357, 361]: We explore establishing rich semantic connections between language and image/video data, with a focus on Multimodal Summarization with Multimodal Output (MSMO) task. By aligning the semantic content of language with visual elements, the resulting models can possess a more nuanced understanding of the underlying concepts.
- Interpretability [361, 363]: We delve into the application of Optimal Transport-based approaches to learn cross-domain alignment, enabling models to provide interpretable explanations of their multimodal reasoning process.
- Language grounding in robot learning [355]: This research aims to develop techniques for learning executable plans from visual observations by incorporating latent language encoding. Models are trained to understand and interpret visual cues while leveraging the rich semantic information encoded in language.
- Retrieval-augmented Multimodal LLM [353]: We develop a retrieval-augmented Multimodal LLM model, which is capable of recognizing and providing knowledgeable answers in real-world entity-centric Visual Question Answering (VQA).
- ECG-to-text generation [349]: We bridge the gap by transferring the knowledge of LLMs to clinical ECG for diagnosis report generation and zero-shot disease detection.
- Connection between human language and brain signals [146]: We explore the relationship and dependency between EEG and human language to reveal the inner connection.
- Clinical retrieval system for Cardiovascular Magnetic Resonance (CMR) Imaging [351]: We design a retrieval system that can automatically match the input signal to the most similar records in the database. This functionality can significantly aid in diagnosing diseases and reduce physicians' workload.

Datasets and Benchmark Our work has focused on:

- Robustness evaluation benchmark of multimodal models [363]: We develop comprehensive evaluation metrics and methodologies to assess the robustness of multimodal models. By simulating distribution shifts and measuring the model's performance under different scenarios, we can gain a deeper understanding of the model's adaptability and identify potential vulnerabilities.
- New MSMO dataset [357]: We propose a new dataset named MMSum to solve the problems within existing MSMO datasets, such as insufficient maintenance, data inaccessibility, limited size, and categorization, etc., spanning 17 principal categories and 170 subcategories.
- New Livestream video dataset [347]: We introduce a new large dataset of Livestream videos, which contains 11,285 Livestream videos with a total duration of 15,038.4 hours.
- New CMR dataset [351]: The existing work falls short in providing a large CMR dataset, we take the initiative to gather a comprehensive dataset consisting of 13,787 studies derived from actual clinical cases.
- New entity-centric VQA dataset [353]: We have developed the SnapNTell dataset, distinct from traditional VQA datasets as (1) It encompasses a wide range of categorized entities,

each represented by images and explicitly named in the answers; (2) It features QA pairs that require extensive knowledge for accurate responses. The dataset is organized into 22 major categories, containing 7,568 unique entities in total.

Applications The works in this thesis are listed in Table 1.1 based on the topics.

Table 1.1: Thesis topics.

	Model/Algorithm	Dataset/Benchmark	Multimedia	Application Robotics	Healthcare	Venue
Alignment	SCCS LiveSeg MMSum	LiveSeg MMSum Entity6K	SCCS LiveSeg MMSum Entity6K			ACL Findings 2023 [361] WACV 2023 [347] CVPR 2024 [357] Under Review [348]
Robustness	MMRobustness Cardiac-MT Interp-OT GeoECG	MMRobustness	Interp-OT		Cardiac-MT GeoECG	DMLR 2024 [363] ICASSP 2023 [362] ICML 2023 [574] PMLR MLHC 2022 [575]
Generalization	SUM+APM ECG-LLM MTAM CMRformer ECG-Encoding SnapNTell	CMRformer SnapNTell	SnapNTell	SUM+APM	ECG-LLM MTAM CMRformer ECG-Encoding	NAACL 2024 [355] EACL Findings 2023 [349] EMNLP Findings 2023 [146] ICML 2023 Workshop [351] ML4H 2023 [358] Under review [353]

The work presented in this thesis has already had a significant impact, which we categorize into 1) Application impact, 2) Academic impact, and 3) Impact through code release.

Application Impact

- [347] has been patented by Adobe for production on Behance Livestream [197].
- [351] has been implemented by Cleveland Clinic for clinical trials.

Academic Impact The work presented here has been previously published in top-tier outlets in different venues, as in Table 12.1, especially:

- [363] was accepted as the very **first** paper of the Journal of Data-centric Machine Learning Research (DMLR).
- [357] was accepted as **Poster (Highlight)** in CVPR 2024, which is **Top 11.9%** among all accepted papers.
- [355] was accepted as a **spotlight** for the ICML 2023 Workshop on Interactive Learning with Implicit Human Feedback.

Impact Through Code Release To enable the reproducibility and extendibility of our work, we have publicly released the source code for the algorithms presented in this thesis. As of January 28, 2024:

- [363] has 30 stars, 18 clones, 173 viewers, and 17 citations (as of March 24, 2024).
- [357] has 24 stars, 16 clones, and 424 viewers (as of March 24, 2024).
- [267] has 141 citations.

Other Contributions Here we list some other contributions that happened during the graduate study but are not introduced in the thesis.

- **Data augmentation:** Data augmentation techniques can improve the models’ robustness. In [362, 458, 575], we synthesize diverse examples encompassing a wide range of data distributions; models can learn to generalize better and exhibit improved performance in novel scenarios. We propose a physiologically-inspired data augmentation method to improve the performance, generalization, and robustness of the ECG prediction model, where the new data augmentation method was proposed from a probability perspective. We perturb the data distribution towards other classes along the geodesic in a Wasserstein space. Also, the ground metric of this Wasserstein space is computed via a set of physiological features so that the perturbation lies on a manifold that exploits the physiological properties.
- **ECG-encoding:** In [358], we encode ECG as images and adopted a vision-language learning paradigm to jointly learn vision-language alignment between encoded ECG images and ECG diagnosis reports. Encoding ECG into images can result in an efficient ECG retrieval system, which can be highly practical and useful in clinical applications.
- **Geometry-aware representations with Wasserstein distance:** In [574], we learn diverse representations based on the Wasserstein distance. In particular, by choosing the distance metric in the primal definition of the Wasserstein distance to reflect our prior knowledge about the data, we can formulate the corresponding dual problem and learn diverse representations, maximizing the pairwise Wasserstein distances under certain model smoothness constraints.

Thesis Statement Multimodal intelligence refers to the ability of a system to integrate and process information from multiple sensory modalities, such as visual, text, tactile inputs, etc., to achieve complex understanding and perform tasks. This type of intelligence is crucial in environments where diverse types of data are present, enabling effective interaction, decision-making, and problem-solving by leveraging the complementary strengths of each modality.

This thesis demonstrates that by enhancing the modeling of patterns across different modalities, we can achieve a more effective utilization of the unique modality equivalence learned through abstract multimodal representations. This improved modeling can lead to advancements in cross-modal applications, increasing the robustness of multimodal models under distribution shifts and enhancing their generalization abilities. Consequently, this thesis aims to advance the field of multimodal AI by focusing on the enhancement of alignment, robustness, and generalizability, ultimately leading to the development of more sophisticated and efficient multimodal AI systems.

However, a notable challenge in this domain is the opaque nature of these complex models. The internal logic behind their alignment across different modalities is often not transparent, posing difficulties in understanding and interpreting their behavior. Future research in this area could delve into the interpretability of multimodal AI systems, exploring methods to elucidate the alignment logic across different modalities and how it can be leveraged more efficiently.

Chapter 2

Background and Preliminary

This chapter reviews the current literature of multimodal learning related work. We review general methodologies and models in the literature. More specific ones will be introduced later in each chapter.

2.1 Background

Multimodal Learning has advanced quickly in recent years with appealing applications in different fields, i.e., embodied learning [38, 178, 192, 294], multimedia image/video and language understanding [99, 179, 389, 580], and psychology [146, 267]. Thanks to the larger datasets [334, 367, 406, 407, 539] and larger transformer models [43, 59, 68, 253, 546], many powerful multimodal image-text models have been developed and shown great capability. However, unlike unimodal models, the robustness study of multimodal models under distribution shift has rarely been explored.

Multimodal Alignment Aligning representations from different modalities is important in multimodal learning. Exploring the explicit relationship across vision and language has drawn significant attention [481]. [461, 514, 538] adopted attention mechanisms, [90] composed pairwise joint representation, [57, 502, 547] learned fine-grained or hierarchical alignment, [231, 503] decomposed the inputs into sub-tokens, [470, 530] adopted graph attention for reasoning, and [141, 367, 465, 522] applied contrastive learning algorithms.

Multimodal Summarization explored multiple modalities for summary generation by learning the alignment [114, 324, 494, 541] learned the relevance or mapping in the latent space between different modalities. In addition to only generating visual summaries, [19, 236, 576] generated textual summaries by taking audio, transcripts, or documents as input along with videos or images, using seq2seq model [445] or attention mechanism [24]. The methods above explored using multiple modalities' information to generate a single modality output, either textual or visual summary. Recent trends on the MSMO task have also drawn much attention [114, 115, 154, 295, 360, 361, 452, 552, 557, 576]. Specifically, [452] summarized a video and text document into a cover frame and a one-sentence summary. The most significant difference between multimodal summarization and MSMO lies in the inclusion of multiple modalities in the output.

Multimodal LLMs Expanding text-only LLMs to interpret visual information typically involves integrating a visual encoder with a frozen LLM, using extensive image captioning datasets for alignment [68, 219, 504]. This integration can be accomplished through methods such as adapter-based tuning [8], which fine-tunes a small portion of the model to process visual inputs, or prefix tuning [463], where trained prefixed vectors are inputted to guide the frozen LLM towards contextually relevant text outputs based on the visual data. These techniques allow LLMs to maintain their linguistic prowess while gaining visual understanding without full model retraining [534].

2.2 Preliminary

Several fundamental methodologies have been established in the literature. In this section, we present their preliminaries, which will be utilized in subsequent chapters.

Optimal Transport (OT) Basis OT is the problem of transporting mass between two discrete distributions supported on latent feature space \mathcal{X} . Let $\boldsymbol{\mu} = \{\mathbf{x}_i, \mu_i\}_{i=1}^n$ and $\boldsymbol{\nu} = \{\mathbf{y}_j, \nu_j\}_{j=1}^m$ be the discrete distributions of interest, where $\mathbf{x}_i, \mathbf{y}_j \in \mathcal{X}$ denotes the spatial locations and μ_i, ν_j , respectively, denoting the non-negative masses. Without loss of generality, we assume $\sum_i \mu_i = \sum_j \nu_j = 1$. $\pi \in \mathbb{R}_+^{n \times m}$ is a valid transport plan if its row and column marginals match $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$, respectively, which is $\sum_i \pi_{ij} = \nu_j$ and $\sum_j \pi_{ij} = \mu_i$. Intuitively, π transports π_{ij} units of mass at location \mathbf{x}_i to new location \mathbf{y}_j . Such transport plans are not unique, and one often seeks a solution $\pi^* \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu})$ that is most preferable in other ways, where $\Pi(\boldsymbol{\mu}, \boldsymbol{\nu})$ denotes the set of all viable transport plans. OT finds a solution that is most cost-effective w.r.t. cost function $C(\mathbf{x}, \mathbf{y})$:

$$\mathcal{D}(\boldsymbol{\mu}, \boldsymbol{\nu}) = \sum_{ij} \pi_{ij}^* C(\mathbf{x}_i, \mathbf{y}_j) = \inf_{\pi \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu})} \sum_{ij} \pi_{ij} C(\mathbf{x}_i, \mathbf{y}_j) \quad (2.1)$$

where $\mathcal{D}(\boldsymbol{\mu}, \boldsymbol{\nu})$ is known as OT distance. $\mathcal{D}(\boldsymbol{\mu}, \boldsymbol{\nu})$ minimizes the transport cost from $\boldsymbol{\mu}$ to $\boldsymbol{\nu}$ w.r.t. $C(\mathbf{x}, \mathbf{y})$. When $C(\mathbf{x}, \mathbf{y})$ defines a distance metric on \mathcal{X} , and $\mathcal{D}(\boldsymbol{\mu}, \boldsymbol{\nu})$ induces a distance metric on the space of probability distributions supported on \mathcal{X} , it becomes the Wasserstein Distance (WD).

Wasserstein Distance As above, Wasserstein distance (WD) is introduced in OT, which is a natural type of divergence for registration problems as it accounts for the underlying geometry of the space, and has been used for multimodal data matching and alignment tasks [54, 81, 230, 360, 540, 575]. In Euclidean settings, OT introduces WD $\mathcal{W}(\mu, \nu)$, which measures the minimum effort required to “displace” points across measures μ and ν , where μ and ν are values observed in the empirical distribution. In our setting, we compute the temporal-pairwise Wasserstein Distance on EEG features and language features, which are $(\mu, \nu) = (V_e, V_t)$. For simplicity without loss of generality, assume $\mu \in P(\mathbb{X})$ and $\nu \in P(\mathbb{Y})$ denote the two discrete distributions, formulated as $\mu = \sum_{i=1}^n u_i \delta_{x_i}$ and $\nu = \sum_{j=1}^m v_j \delta_{y_j}$, with δ_x as the Dirac function centered on x . $\Pi(\mu, \nu)$ denotes all the joint distributions $\gamma(x, y)$, with marginals $\mu(x)$ and $\nu(y)$. The weight vectors $u = \{u_i\}_{i=1}^n \in \Delta_n$ and $v = \{v_i\}_{i=1}^m \in \Delta_m$ belong to the n - and m -dimensional simplex, respectively. The WD between the two discrete distributions μ and ν is defined as:

$$\mathcal{WD}(\mu, \nu) = \inf_{\gamma \in \Pi(\mu, \nu)} \mathbb{E}_{(x, y) \sim \gamma} [c(x, y)] = \min_{\mathbf{T} \in \Pi(\mathbf{u}, \mathbf{v})} \sum_{i=1}^n \sum_{j=1}^m T_{ij} \cdot c(x_i, y_j) \quad (2.2)$$

where $\Pi(u,v)=\{T \in \mathbb{R}_+^{n \times m} | T \mathbf{1}_m = u, T^\top \mathbf{1}_n = v\}$, $\mathbf{1}_n$ denotes an n -dimensional all-one vector, and $c(x_i, y_j)$ is the cost function evaluating the distance between x_i and y_j .

Canonical Correlation Analysis Canonical Correlation Analysis (CCA) is a method for exploring the relationships between two multivariate sets of variables. It learns the linear transformation of two vectors to maximize the correlation between them, which is used in many multimodal problems [16, 129, 352]. In this chapter, we apply CCA to capture the cross-domain relationship. Let low-level transformed EEG features be V_e and low-level language features be V_t . We assume $(V_e, V_t) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2}$ has covariances $(\Sigma_{11}, \Sigma_{22})$ and cross-covariance Σ_{12} . CCA finds pairs of linear projections of the two views, $(w_1' V_e, w_2' V_t)$ that are maximally correlated:

$$(w_1^*, w_2^*) = \operatorname{argmax}_{w_1, w_2} \operatorname{corr}(w_1' V_e, w_2' V_t) = \operatorname{argmax}_{w_1, w_2} \frac{w_1' \Sigma_{12} w_2}{\sqrt{w_1' \Sigma_{11} w_1 w_2' \Sigma_{22} w_2}} \quad (2.3)$$

Transformer Architecture The transformer is based on the attention mechanism [468]. The original transformer model is composed of an encoder and a decoder. The encoder maps an input sequence into a latent representation, and the decoder uses the representation with other inputs to generate a target sequence.

First, we feed out the input into an embedding layer, which is a learned vector representation of the input feature, by mapping the features to a vector with continuous values. Then we inject positional information into the embeddings by:

$$PE_{(pos, 2i)} = \sin\left(pos/10000^{2i/d_{\text{model}}}\right), PE_{(pos, 2i+1)} = \cos\left(pos/10000^{2i/d_{\text{model}}}\right) \quad (2.4)$$

The attention model contains two sub-modules, a multi-headed attention model and a fully connected network. The multi-headed attention computes the attention weights for the input and produces an output vector with encoded information on how each feature should attend to all other features in the sequence. There are residual connections around each of the two sub-layers followed by a layer normalization, where the residual connection means adding the multi-headed attention output vector to the original positional input embedding, which helps the network train by allowing gradients to flow through the networks directly.

Multi-headed attention applies a self-attention mechanism, where the input goes into three distinct fully connected layers to create the query, key, and value vectors. The output of the residual connection goes through a layer normalization.

$$\begin{aligned} \text{MultiHead}(Q, K, V) &= \text{Concat}(\text{head}_1, \dots, \text{head}_h) W^O \\ \text{where head}_i &= \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \end{aligned} \quad (2.5)$$

The attention model contains N same layers, and each layer contains two sub-layers, which are a multi-head self-attention model and a fully connected feed-forward network. Residual connection and normalization are added in each sub-layer. So the output of the sub-layer can be expressed as: $\text{Output} = \text{LayerNorm}(x + (\text{SubLayer}(x)))$ For the Multi-head self-attention module, the attention can be expressed as: $\text{attention} = \text{Attention}(Q, K, V)$, where multi-head attention uses h different linear transformations to project query, key, and value, which are Q , K , and V , respectively, and finally concatenate different attention results:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h) W^O \quad (2.6)$$

$$head_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \quad (2.7)$$

where the projections are parameter matrices:

$$W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}, \quad W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}, \quad W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v}, \quad W_i^O \in \mathbb{R}^{d_v \times d_{\text{model}}} \quad (2.8)$$

where the computation of attention adopted scaled dot-product:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2.9)$$

Vision Transformer (ViT) A Vision Transformer (ViT) [92] is a type of transformer specifically developed for computer vision tasks. Unlike text-based transformers that split text into tokens, a ViT divides an input image into several patches. Each patch is then converted into a vector and reduced to a smaller dimension through matrix multiplication. These vector embeddings are processed by a transformer encoder, similar to how token embeddings are handled. ViTs are utilized in various applications, including image recognition, image segmentation, and so on.

Given an input image I with dimensions $H \times W \times C$, where H is the height, W is the width, and C is the number of channels (e.g., 3 for RGB): The image is divided into N patches, each of size $P \times P$, resulting in patches with dimensions $\frac{H}{P} \times \frac{W}{P} \times C$. Each patch is then flattened and linearly projected to a D -dimensional embedding vector using a learnable matrix $E \in \mathbb{R}^{(P^2 \cdot C) \times D}$. Positional embedding is then added to the patch embeddings to retain positional information. The final input embeddings for the transformer encoder is the sum of patch embedding and position embedding. The input embeddings are passed through a series of transformer encoder layers, each consisting of multi-head self-attention (MHSA) and feed-forward (FF) networks. The output of the transformer encoder is then used for various vision tasks, such as classification, segmentation, etc.

CLIP CLIP [367] is designed to understand and generate associations between textual and visual data. The model is trained on a large dataset of images and their corresponding textual descriptions, allowing it to learn a wide range of visual concepts and their linguistic representations.

CLIP consists of two encoders: an image encoder and a text encoder. The goal of CLIP is to learn a joint embedding space where corresponding images and texts are close to each other. Let I be an input image. The image encoder f_I maps I to a D -dimensional embedding vector:

$$\mathbf{v} = f_I(I) \in \mathbb{R}^D \quad (2.10)$$

Let T be an input text (e.g., a caption or a sentence). The text encoder f_T maps T to a D -dimensional embedding vector:

$$\mathbf{w} = f_T(T) \in \mathbb{R}^D \quad (2.11)$$

CLIP uses a contrastive loss to train the encoders. For a batch of N image-text pairs, the contrastive loss is defined as:

$$\mathcal{L} = - \sum_{i=1}^N \log \frac{\exp(\mathbf{v}_i \cdot \mathbf{w}_i / \tau)}{\sum_{j=1}^N \exp(\mathbf{v}_i \cdot \mathbf{w}_j / \tau)} \quad (2.12)$$

where τ is a temperature parameter, and $\mathbf{v}_i \cdot \mathbf{w}_j$ is the dot product between the image embedding \mathbf{v}_i and the text embedding \mathbf{w}_j . The goal of the training is to minimize the contrastive loss, which encourages the model to align the embeddings of corresponding image-text pairs while pushing apart the embeddings of non-corresponding pairs.

Part I

Learning Cross-modal Semantic Alignment

Chapter 3

Multimodal Summarization via Cross-domain Alignment

In this chapter, we start with the discussion of learning cross-domain alignment, with a focus on a new multimedia application named Multimodal summarization with multimodal output (**MSMO**). MSMO is a recently explored application in language grounding. It plays an essential role in real-world applications, i.e., automatically generating cover images and titles for news articles or providing introductions to online videos. However, existing methods extract features from the whole video and article and use fusion methods to select the representative one, thus usually ignoring the critical structure and varying semantics with video/document. In this chapter, we propose a Semantics-Consistent Cross-domain Summarization (SCCS) model based on optimal transport alignment with visual and textual segmentation. Our method first decomposes both videos and articles into segments in order to capture the structural semantics, and then follows a cross-domain alignment objective with optimal transport distance, which leverages multimodal interaction to match and select the visual and textual summary. We evaluate our method on three MSMO datasets, and achieved performance improvement by 8% & 6% of textual and 6.6% & 5.7% of video summarization, respectively, which demonstrated the effectiveness of our method in producing high-quality multimodal summaries.

3.1 Introduction

New multimedia content in the form of short videos and corresponding text articles has become a significant trend in influential digital media. This popular media type has been shown to be successful in drawing users' attention and delivering essential information in an efficient manner. **MSMO** has recently drawn increasing attention. Different from traditional video or textual summarization [143, 191], where the generated summary is either a keyframe or textual description, MSMO aims at producing both visual and textual summaries simultaneously, making this task more complicated. Previous works addressed the MSMO task by processing the whole video and the whole article together which overlooked the structure and semantics of different domains [96, 114, 115, 150, 295, 399, 576].

The video and article can be regarded as being composed of several topics related to the main idea, while each topic specifically corresponds to one sub-idea. Thus, treating the whole

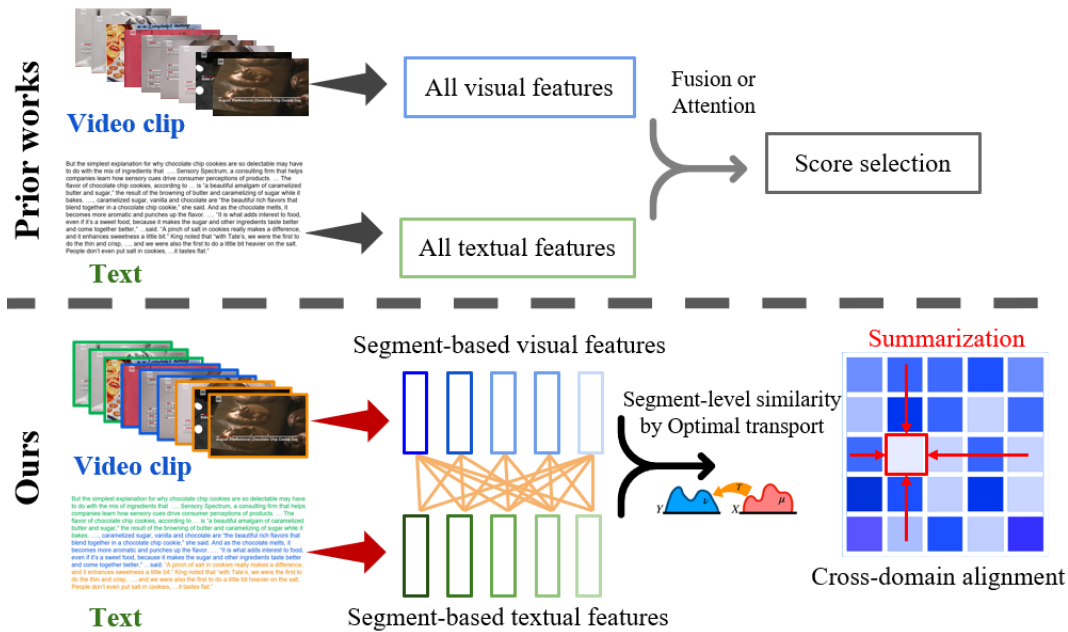


Figure 3.1: Comparison with previous work: We proposed a segment-level cross-domain alignment model to preserve the structural semantics consistency within two domains for the MSMO task. We solve an optimal transport problem to optimize the cross-domain distance, which in turn finds the optimal match.

video or article uniformly and learning a general representation ignores these structural semantics and easily leads to biased summarization. To address this problem, instead of learning averaged representations for the whole video & article, we focus on exploiting the original underlying structure. The comparison of our approach and previous works is illustrated in Figure 3.1. Our model first decomposes the video & article into segments to discover the content structure, then explores the cross-domain semantics relationship at the segment level. We believe this is a promising approach to exploit the *consistency* lie in the structural semantics between different domains.

Previous models applied attention or fusion mechanisms to compute image-text relevance scores, finding the best match of the sentences/images within the whole document/video, regardless of the context, which used one domain as an anchor. However, an outstanding anchor has more weight in selecting the corresponding pair. To overcome this, we believe the semantics structure is a crucial characteristic that can not be ignored. Based on this hypothesis, we propose Semantics-Consistent Cross-domain Summarization (SCCS), which explores segment-level cross-domain representations through OT-based multimodal alignment to generate both visual and textual summaries. We decompose the video/document into segments based on its semantic structure, then generate sub-summaries of each segment as candidates. We select the final summary from these candidates instead of a global search, so all candidates are in a fair competition arena.

Our contributions can be summarized as follows :

- We propose SCCS, a segment-level alignment model for MSMO tasks.
- Our method preserves the structural semantics and explores the cross-domain relationship through optimal transport to match and select the visual and textual summary.
- On three datasets, our method outperforms baselines in both textual and video summarization results qualitatively and quantitatively.



Figure 3.2: SCCS at work: A real example of the summarization process given by our SCCS method. Here we conduct OT-based cross-domain alignment to each keyframe-sentence pair, and a smaller OT distance means better alignment. (For example, the best-aligned text and image summary (0.08) delivers the flooding content clearly and comprehensively.)

- Our method serves as a hierarchical MSMO framework and provides better interpretability via OT alignment. The OT coupling shows sparse patterns and specific temporal structure for the embedding vectors of ground-truth-matched video and text segments, providing interpretable learned representations.

Since MSMO generates both visual & textual summaries, We believe the optimal summary comes from the video and text pair that are both 1) semantically consistent, and 2) best matched globally in a cross-domain fashion. In addition, our framework is more computationally efficient as it conducts cross-domain alignment at the segment level instead of inputting whole videos/articles.

3.2 Related Work

Multimodal Summarization Multimodal summarization explored multiple modalities, i.e., audio signals, video captions, transcripts, video titles, etc, for summary generation. [114, 324, 494, 541] learned the relevance or mapping in the latent space between different modalities. In addition to only generating visual summaries, [19, 236, 576] generated textual summaries by taking audio, transcripts, or documents as input along with videos or images, using seq2seq model [445] or attention mechanism [24]. Recent trending on the MSMO task has also drawn much attention [114, 115, 295, 552, 576].

Optimal Transport OT studies the geometry of probability spaces [471], a formalism for finding and quantifying mass movement from one probability distribution to another. OT defines the Wasserstein metric between probability distributions, revealing a canonical geometric structure with rich properties to be exploited. The earliest contribution to OT originated from Monge in the eighteenth century. Kantorovich rediscovered it under a different formalism, namely the Linear Programming formulation of OT. With the development of scalable solvers, OT is widely applied to many real-world problems and applications [11, 54, 55, 96, 105, 215, 230, 540, 573].

3.3 Proposed Method

SCCS is a segment-level cross-domain semantics alignment model for the MSMO task, where MSMO aims at generating both visual and language summaries. We follow the problem setting in

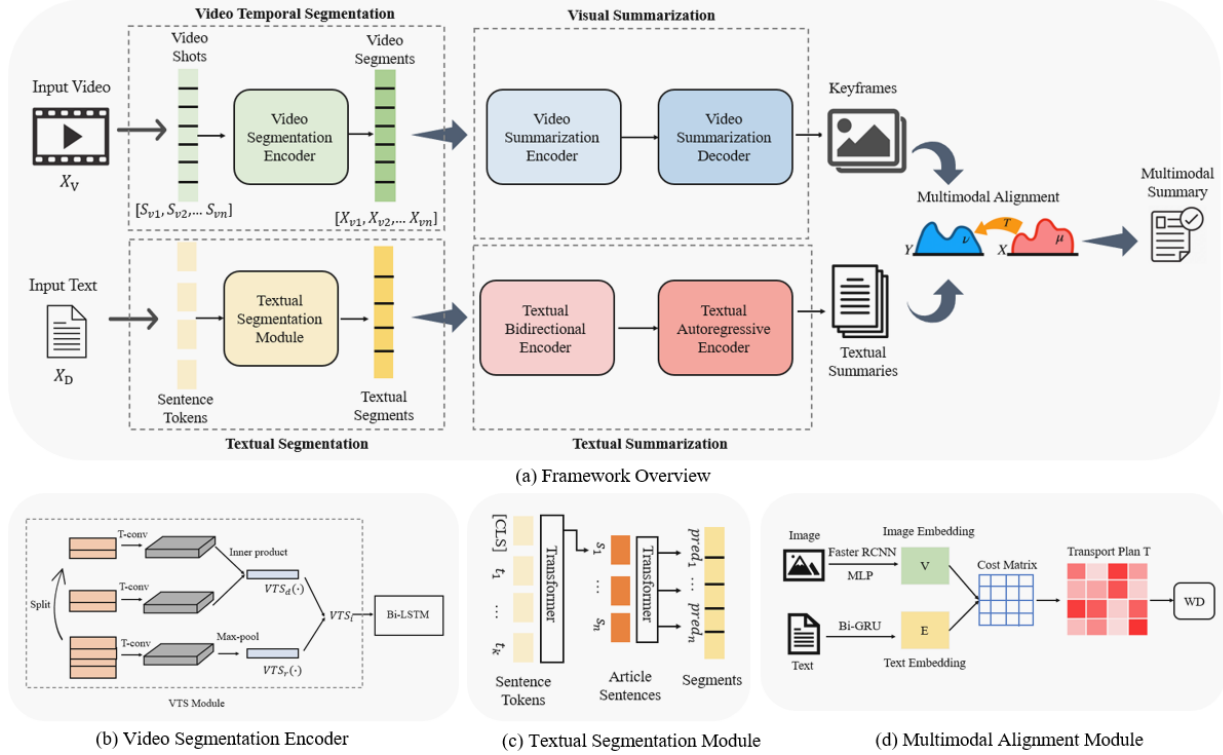


Figure 3.3: SCCS framework: (a) The computational framework of the SCCS model, which takes multimodal inputs (videos & text documents) and generates multimodal summaries. The framework includes five modules: video temporal segmentation, visual summarization, textual segmentation, textual summarization, and multimodal alignment. (b) The structure of the video segmentation encoder. (c) The architecture of the textual segmentation module. (d) The multimodal alignment module for multimodal summaries.

[295], for a multimedia source with documents and videos, the document $X_D = \{x_1, x_2, \dots, x_d\}$ has d words, and the ground truth textual summary $Y_D = \{y_1, y_2, \dots, y_g\}$ has g words. A corresponding video X_V is associated with the document in pair, and there exists a ground truth cover picture Y_V that can represent the most important information to describe the video. Our SCCS model generates both textual summaries Y'_D and video keyframes Y'_V .

SCCS consists of five modules, as shown in Figure 3.3(a): video temporal segmentation (Section 3.3.1), visual summarization (Section 3.3.3), textual segmentation (Section 3.3.2), textual summarization (Section 3.3.4), and cross-domain alignment (Section 3.3.5). Each module will be introduced in the following subsections.

3.3.1 Video Temporal Segmentation

Video temporal segmentation splits the original video into small segments, which summarization tasks build upon. The segmentation is formulated as a binary classification problem on the segment boundaries, similar to [375]. For a video X_V , the video segmentation encoder separates the video sequence into segments $[X_{v1}, X_{v2}, \dots, X_{vm}]$, where n is the number of segments.

As shown in Figure 3.3(b), the video segmentation encoder contains a VTS module and a

Bi-LSTM [134]. Video X_V is first split into shots $[S_{v1}, S_{v2}, \dots, S_{vn}]$ [47], then the VTS module takes a clip of the video with $2\omega_b$ shots as input and outputs a boundary representation b_i . The boundary representation captures both differences and relations between the shots before and after. VTS consists of two branches, VTS_d and VTS_r , as shown in Equation 3.1.

$$\begin{aligned} b_i &= \text{VTS} \left([S_{vi-(\omega_b-1)}, \dots, S_{vi+\omega_b}] \right) \\ &= \begin{bmatrix} \text{VTS}_d \left([S_{vi-(\omega_b-1)}, \dots, P_{vi}], [S_{v(i+1)}, \dots, S_{vi+\omega_b}] \right) \\ \text{VTS}_r \left([S_{vi-(\omega_b-1)}, \dots, P_{vi}, S_{v(i+1)}, \dots, S_{vi+\omega_b}] \right) \end{bmatrix} \end{aligned} \quad (3.1)$$

VTS_d is modeled by two temporal convolution layers, each of which embeds the w_b shots before and after the boundary, respectively, following an inner product operation to calculate the differences. VTS_r contains a temporal convolution layer followed by a max pooling, aiming at capturing the relations of the shots. It predicts a sequence binary labels $[p_{v1}, p_{v2}, \dots, p_{vn}]$ based on the sequence of representatives $[b_1, b_2, \dots, b_n]$. A Bi-LSTM [134] is used with stride $\omega_t/2$ shots to predict a sequence of coarse score $[s_1, s_2, \dots, s_n]$, as shown in Equation 3.2,

$$[s_1, s_2, \dots, s_n] = \text{Bi-LSTM} ([b_1, b_2, \dots, b_n]) \quad (3.2)$$

where $s_i \in [0, 1]$ is the probability of a shot boundary being a scene boundary. The coarse prediction $\hat{p}_{vi} \in \{0, 1\}$ indicates whether the i -th shot boundary is a scene boundary by binarizing s_i with a threshold τ , $\hat{p}_{vi} = \begin{cases} 1 & \text{if } s_i > \tau \\ 0 & \text{otherwise} \end{cases}$. The results with $\hat{p}_{vi} = 1$ result in the learned video segments $[X_{v1}, X_{v2}, \dots, X_{vm}]$.

3.3.2 Textual Segmentation

The textual segmentation module takes the whole document or articles as input and splits the original input into segments based on context understanding. We used a hierarchical BERT as the textual segmentation module [275], which is the current state-of-the-art method. As shown in Figure 3.3(c), the textual segmentation module contains two-level transformer encoders, where the first-level encoder is for sentence-level encoding, and the second-level encoder is for article-level encoding. The hierarchical BERT starts by encoding each sentence with $\text{BERT}_{\text{LARGE}}$ independently, and then the tensors produced for each sentence are fed into another transformer encoder to capture the representation of the sequence of sentences. All the sequences start with a [CLS] token to encode each sentence with BERT at the first level. If the segmentation decision is made at the sentence level, we use the [CLS] token as input for the second-level encoder. The [CLS] token representations from sentences are passed into the article encoder, which can relate the different sentences through cross-attention.

3.3.3 Visual Summarization

The visual summarization module generates visual keyframes from each video segment as its corresponding summary. We use an encoder-decoder architecture with attention as the visual summarization module [193], taking each video segment as input and outputting a sequence of keyframes. The encoder is a Bi-LSTM [134] to model the temporal relationship of video frames,

where the input is $X = [x_1, x_2, \dots, x_T]$ and the encoded representation is $E = [e_1, e_2, \dots, e_T]$. The decoder is a LSTM [164] to generate output sequences $D = [d_1, d_2, \dots, d_m]$. To exploit the temporal ordering across the entire video, an attention mechanism is used: $E_t = \sum_{i=1}^m \alpha_t^i e_i$, s.t. $\sum_{i=1}^m \alpha_t^i = 1$. Similar in [164], the decoder function can be written as:

$$\left[\begin{array}{c} p(d_t | \{d_i | i < t\}, E_t) \\ s_t \end{array} \right] = \psi(s_{t-1}, d_{t-1}, E_t) \quad (3.3)$$

where s_t is the hidden state, E_t is the attention vector at time t , α_t^i is the attention weight between the inputs and the encoder vector, ψ is the decoder function (LSTM). To obtain α_t^i , the relevance score γ_t^i is computed by $\gamma_t^i = \text{score}(s_{t-1}, e_i)$, where the score function decides the relationship between the i -th visual features e_i and the output scores at time t : $\gamma_t^i = e_i^T W_a s_{t-1}$, $\alpha_t^i = \exp(\gamma_t^i) / \sum_{j=1}^m \exp(\gamma_t^j)$.

3.3.4 Textual Summarization

Language summarization can produce a concise and fluent summary which should preserve the critical information and overall meaning. Our textual summarization module takes BART [235] as the summarization model to generate abstractive textual summary candidates. BART is a denoising autoencoder that maps a corrupted document to the original document it was derived from. As in Figure 3.3(a), BART is an encoder-decoder Transformer pre-trained with a denoising objective on text. We take the fine-tuned BART on CNN and Daily Mail datasets for the summarization task [308, 411].

3.3.5 Cross-Domain Alignment via OT

Our cross-domain alignment (CDA) module learns the alignment between keyframes and textual summaries to generate the final multimodal summaries. Our alignment module is based on OT, which has been explored in several cross-domain tasks [54, 274, 540].

Our cross-domain alignment (CDA) module As shown in Figure 3.3(d), in CDA, the image features $\mathbf{V} = \{\mathbf{v}_k\}_{k=1}^K$ are extracted from pre-trained ResNet-101 [156] concatenated to faster R-CNN [384] as [540], where an image can be represented as a set of detected objects, each associated with a feature vector. For text features, every word is embedded as a feature vector and processed by a Bi-GRU [65] to account for context [540]. The extracted image and text embeddings are $\mathbf{V} = \{\mathbf{v}_i\}_1^K$, $\mathbf{E} = \{e_i\}_1^M$, respectively.

As in [540], we take image and text sequence embeddings as two discrete distributions supported on the same feature representation space. OT formulation has been introduced in Chapter 2. Solving an OT transport plan between the two naturally constitutes a matching scheme to relate cross-domain entities [540]. To evaluate the OT distance, we compute a pairwise similarity between \mathbf{V} and \mathbf{E} using cosine distance:

$$C_{km} = C(e_k, v_m) = 1 - \frac{\mathbf{e}_k^T \mathbf{v}_m}{\|\mathbf{e}_k\| \|\mathbf{v}_m\|} \quad (3.4)$$

Then the OT can be formulated as:

$$\mathcal{L}_{\text{OT}}(\mathbf{V}, \mathbf{E}) = \min_{\mathbf{T}} \sum_{k=1}^K \sum_{m=1}^M \mathbf{T}_{km} C_{km} \quad (3.5)$$

Algorithm 1 Compute Alignment Distance

- 1: **Input:** $\mathbf{E} = \{\mathbf{e}_i\}_1^M$, $\mathbf{V} = \{\mathbf{v}_i\}_1^K$, β
 - 2: $\mathbf{C} = C(\mathbf{V}, \mathbf{E})$, $\sigma \leftarrow \frac{1}{m} \mathbf{1}_m$, $\mathbf{T}^{(1)} \leftarrow \mathbf{1}\mathbf{1}^T$
 - 3: $\mathbf{G}_{ij} \leftarrow \exp\left(-\frac{C_{ij}}{\beta}\right)$
 - 4: **for** $t = 1, 2, 3, \dots, N$ **do**
 - 5: $\mathbf{Q} \leftarrow \mathbf{G} \odot \mathbf{T}^{(t)}$
 - 6: $\delta \leftarrow \frac{1}{K\mathbf{Q}\sigma}$, $\sigma \leftarrow \frac{1}{M\mathbf{Q}^T\delta}$
 - 7: $\mathbf{T}^{(t+1)} \leftarrow \text{diag}(\delta)\mathbf{Q}\text{diag}(\sigma)$
 - 8: **end for**
 - 9: $\text{Dis} = \langle C^T, T \rangle$
-

where $\sum_m \mathbf{T}_{km} = \mu_k$, $\sum_k \mathbf{T}_{km} = v_m$, $\forall k \in [1, K]$, $m \in [1, M]$, $\mathbf{T} \in \mathbb{R}_+^{K \times M}$ is the transport matrix, d_k and d_m are the weight of \mathbf{v}_k and \mathbf{e}_m in a given image and text sequence, respectively. We assume the weight for different features to be uniform, i.e., $\mu_k = \frac{1}{K}$, $v_m = \frac{1}{M}$. The objective of optimal transport involves solving linear programming and may cause potential computational burdens since it has $O(n^3)$ efficiency. To solve this issue, we add an entropic regularization term equation (3.5), and the objective of our optimal transport distance becomes:

$$\mathcal{L}_{\text{OT}}(\mathbf{V}, \mathbf{E}) = \min_{\mathbf{T}} \sum_{k=1}^K \sum_{m=1}^M \mathbf{T}_{km} C_{km} + \lambda H(\mathbf{T}) \quad (3.6)$$

where $H(\mathbf{T}) = \sum_{i,j} \mathbf{T}_{i,j} \log \mathbf{T}_{i,j}$ is the entropy, and λ is the hyperparameter that balances the effect of the entropy term. Thus, we are able to apply the celebrated Sinkhorn algorithm [76] to efficiently solve the above equation in $O(n \log n)$. The optimal transport distance computed via the Sinkhorn algorithm is differentiable and can be implemented by [105]. The algorithm is shown in Algorithm 1, where β is a hyper-parameter, \mathbf{C} is the cost matrix, \odot is Hadamard product, $\langle \cdot, \cdot \rangle$ is Frobenius dot-product, matrices are in bold, the rest are scalars.

3.3.6 Multimodal Summaries

During training the alignment module, the **WD** between each keyframe-sentence pair of all the visual & textual summary candidates is computed, where the best match is selected as the final multimodal summaries.

3.4 Datasets and Baselines

Datasets We evaluated our models on three datasets: the VMSMO dataset, Daily Mail dataset, and CNN dataset from [114, 115, 295]. The VMSMO dataset contains 184,920 samples, including articles and corresponding videos. Each sample is assigned with a textual summary and a video with a cover picture. We adopt the available data samples from [295]. The Daily Mail dataset contains 1,970 samples, and the CNN dataset contains 203 samples, which include video titles, images, and captions, similar to [162]. For data splitting, we take the same experimental setup as [295] for the

VMSMO dataset. For the Daily Mail dataset and CNN dataset, we split the data by 70%, 10%, and 20% for train, validation, and test sets, respectively, same as [114, 115].

Baselines We select state-of-the-art MSMO baselines and representative pure video & textual summarization baselines for comparison. For the VMSMO dataset, we compare our method with (i) multimodal summarization baselines (MSMO, MOF [576, 577], and DIMS [295]), (ii) video summarization baselines (Synergistic [139] and PSAC [249]), and (iii) textual summarization baselines (Lead [411], TextRank [291], PG [411], Unified [176], and GPG [419]). For Daily Mail and CNN datasets, we compare our method with (i) multimodal baselines (MSMO [576], Img+Trans [172], TFN [542], HNNattTI [52], and M²SM [114, 115]), (ii) video summarization baselines (VSUMM [80] and DR-DSN [566]), and (iii) textual summarization baselines (Lead3 [411], NN-SE [63], BART [235], T5 [369], and Pegasus [550]).

3.5 Experiments

Experimental Setting and Implementation For the VTS module, we use the same model setting as [47, 375] and the same data splitting setting as [114, 115, 295] in the training process.

The visual summarization model is pre-trained on the TVSum [431] and SumMe [143] datasets. TVSum dataset contains 50 edited videos downloaded from YouTube in 10 categories, and the SumMe dataset consists of 25 raw videos recording various events. Frame-level importance scores for each video are provided for both datasets and used as ground-truth labels. The input visual features are extracted from pre-trained GoogLeNet on ImageNet, where the output of the pool5 layer is used as visual features.

For the textual segmentation module, due to the quadratic computational cost of transformers, we reduce the BERT’s inputs to 64-word pieces per sentence and 128 sentences per document as [275]. We use 12 layers for both the sentence and the article encoders, for a total of 24 layers. In order to use the BERT_{BASE} checkpoint, we use 12 attention heads and 768-dimensional word-piece embeddings. The hierarchical BERT model is pre-trained on the Wiki-727K dataset [222], which contains 727 thousand articles from a snapshot of the English Wikipedia. We used the same data splitting method as [222].

For textual summarization, we adopted the pretrained BART model from [235], which contains 1024 hidden layers and 406M parameters and has been fine-tuned using CNN and Daily Mail datasets. In the cross-domain alignment module, the feature extraction and alignment module is pretrained by MS COCO dataset [258] on the image-text matching task. We added the OT loss as a regularization term to the original matching loss to align the image and text more explicitly.

Evaluation Metrics The quality of the generated textual summary is evaluated by standard Rouge F1 [257] following previous works [60, 295, 411]. ROUGE-1 (R-1), ROUGE-2 (R-2), and ROUGE-L (R-L) refer to the overlap of unigram, bigrams, and the longest common subsequence between the decoded summary and the reference, respectively [257]. Due to the limitation of ROUGE, we also adopt BertScore [554] for evaluation. For the VMSMO dataset, the quality of the chosen cover frame is evaluated by mean average precision (MAP) and recall at position ($R_n@k$) [454, 571], where ($R_n@k$) measures if the positive sample is ranked in the top k positions of n candidates. For

Table 3.1: Comparison with multimodal baselines on the VMSMO dataset. The absolute performance comparison with the baseline MSMO method is marked in red (better) and blue (worse).

Category	Methods	Textual			Video			
		R-1	R-2	R-L	MAP	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$
Video	Synergistic	–	–	–	0.558	0.444	0.557	0.759
	PSAC	–	–	–	0.524	0.363	0.481	0.730
Textual	Lead	16.2	5.3	13.9	–	–	–	–
	TextRank	13.7	4.0	12.5	–	–	–	–
	PG	19.4	6.8	17.4	–	–	–	–
	Unified	23.0	6.0	20.9	–	–	–	–
	GPG	20.1	4.5	17.3	–	–	–	–
Multimodal	<u>MSMO</u>	20.1	4.6	17.3	0.554	0.361	0.551	0.820
	MOF	21.3 (↑ 0.8)	5.7 (↑ 1.1)	17.9 (↑ 0.6)	0.615 (↑ 0.061)	0.455 (↑ 0.094)	0.615 (↑ 0.064)	0.817 (↓ -0.003)
	DIMS	25.1 (↑ 5.0)	9.6 (↑ 5.0)	23.2 (↑ 5.9)	0.654 (↑ 0.100)	0.524 (↑ 0.163)	0.634 (↑ 0.083)	0.824 (↑ 0.004)
Ours	Ours-textual	26.2	9.6	24.1	–	–	–	–
	Ours-video	–	–	–	0.678	0.561	0.642	0.863
	Ours	27.1 (↑ 7.0)	9.8 (↑ 5.2)	25.4 (↑ 8.1)	0.697 (↑ 0.143)	0.582 (↑ 0.221)	0.688 (↑ 0.137)	0.895 (↑ 0.075)

the Daily Mail dataset and CNN dataset, we calculate the cosine image similarity (Cos) between image references and the extracted frames [114, 115].

Results and Discussion The comparison results on the VMSMO dataset of multimodal, video, and textual summarization are shown in Table 3.1. Synergistic and PSAC are pure video summarization approaches, which did not perform as well as multimodal methods, like MOF or DIMS, which means taking additional modality into consideration actually helps to improve the quality of the generated video summaries. Table 3.1 also shows the absolute performance improvement or decrease compared with the MSMO baseline, where the improvements are marked in red and decreases in blue. Overall, our method shows the highest absolute performance improvement than the previous methods on both textual and video summarization results. Our method shows the ability to preserve the structural semantics and is able to learn the alignment between keyframes and textual deceptions, which shows better performance than the previous ones. If comparing the quality of generated textual summaries, our method still outperforms the other multimodal baselines, like MSMO, MOF, DIMS, and also traditional textual summarization methods, like Lead, TextRank, PG, Unified, and GPG, showing the alignment obtained by optimal transport can help to identify the cross-domain inter-relationships.

In Table 3.2, we show the comparison results with multimodal baselines on the Daily Mail and CNN datasets. We can see that for the CNN datasets, our method shows competitive results with Img+Trans, TFN, HNNattTI, and M^2SM on the quality of generated textual summaries. While on the Daily Mail dataset, our approach showed better performance on both textual summaries and visual summaries. We also compare with the traditional pure video summarization baselines and pure textual summarization baselines on the Daily Mail dataset, and the results are shown in Table 3.2. We can find that our approach performed competitive results compared with NN-SE and M^2SM for the quality of the generated textual summary. For visual summarization comparison, we can find that the quality of the generated visual summary by our approach still outperforms the other visual summarization baselines. Still, we also provide absolute performance comparison with baseline MSMO [576], as shown in Table 3.2, our model achieved the highest performance improvement in both Daily Mail and CNN datasets compared with previous baselines. If comparing the quality of generated textual summaries with language model (LM) baselines, our method also

Table 3.2: Comparisons on the Daily Mail and CNN datasets. The absolute performance comparison with the baseline MSMO method is marked in **red** (better) and **blue** (worse).

Category	Methods	CNN dataset				Daily Mail dataset				
		R-1	R-2	R-L	BertScore	R-1	R-2	R-L	BertScore	Cos(%)
Video	VSUMM	-	-	-	-	-	-	-	-	68.74
	DR-DSN	-	-	-	-	-	-	-	-	68.69
Textual	Lead3	-	-	-	-	41.07	17.87	30.90	-	-
	NN-SE	-	-	-	-	41.22	18.15	31.22	-	-
	T5	27.31	8.78	18.22	-	42.32	18.23	33.45	-	-
	Pegasus	27.28	8.83	18.59	-	43.01	18.63	33.54	-	-
	BART	27.50	8.76	18.83	-	42.49	18.67	33.92	-	-
Multimodal	MSMO	26.83	8.11	18.34	12.13	35.38	14.79	25.41	16.25	69.17
	Img+Trans	27.04 (↑ 0.21)	8.29 (↑ 0.18)	18.54 (↑ 0.20)	12.35 (↑ 0.22)	39.28 (↑ 3.90)	16.64 (↑ 1.85)	28.53 (↑ 3.12)	16.43 (↑ 0.18)	-
	TFN	27.68 (↑ 0.85)	8.69 (↑ 0.58)	18.71 (↑ 0.37)	12.59 (↑ 0.46)	39.37 (↑ 3.99)	16.38 (↑ 1.59)	28.09 (↑ 2.68)	16.71 (↑ 0.46)	-
	HNNattTI	27.61 (↑ 0.78)	8.74 (↑ 0.63)	18.64 (↑ 0.30)	12.67 (↑ 0.54)	39.58 (↑ 4.20)	16.71 (↑ 1.92)	29.04 (↑ 3.63)	16.79 (↑ 0.54)	68.76 (↓ -0.41)
	M ² SM	27.81 (↑ 0.98)	8.87 (↑ 0.76)	18.73 (↑ 0.39)	12.72 (↑ 0.59)	41.73 (↑ 6.35)	18.59 (↑ 3.80)	31.68 (↑ 6.27)	16.93 (↑ 0.68)	69.22 (↑ 0.05)
Ours	Ours-textual	-	-	-	12.68	40.28	17.93	31.89	16.98	-
	Ours-video	-	-	-	-	-	-	-	-	70.56
	Ours-Multimodal	28.02 (↑ 1.19)	8.94 (↑ 0.83)	18.89 (↑ 0.55)	13.21 (↑ 1.08)	44.52 (↑ 9.14)	19.87 (↑ 5.08)	35.79 (↑ 10.38)	17.45 (↑ 1.20)	73.19 (↑ 4.02)

Table 3.3: Human evaluation results.

Method	MSMO	TFN	HNNattTI	M ² SM	SCCS
Score	1.84	2.36	3.24	3.4	4.16

outperforms T5, Pegasus, and BART.

Human Evaluation To provide human evaluation results, we asked 5 people (recruited from the institute) to score the results generated by different approaches of CNN and DailyMail datasets. We asked the human judges to score the results of 5 models: MSMO, TFN, HNNattTI, M²SM, and SCCS, as 1-5, where 5 represents the best results. We averaged the voting results from 5 human judges. The performances of 5 models are listed in Table 3.3, showing the result by SCCS is better than the baselines.

Factual Consistency Evaluation Factual consistency is used as another important evaluation criterion for evaluating summarization results [171]. For factual consistency, we adopted the method in [511] and followed the same setting. The same human annotators from Sec 3.5 provided human judgments. We report Pearson correlation coefficient Coe_P here. The results of MSMO, Img+Trans, TFN, HNNattTI, M²SM, and ours, are shown in Table 3.4. In summary, our methods show better results than baselines on factual consistency evaluations.

Ablation Study To evaluate each component’s performance, we performed ablation experiments on different modalities and different datasets. For the VMSMO dataset, we compare the performance of using only visual information, only textual information, and multimodal information. The comparison result is shown in Table 3.1. We also carried out experiments on different modalities using Daily Mail dataset to show the performance of unimodal and multimodal components, and the results are shown in Table 3.2.

For ablation results, when only textual data is available, we adopt BERT [86] to generate text embeddings and K-Means clustering to identify sentences closest to the centroid for textual summary selection. While if only video data is available, we solve the visual summarization task in an unsupervised manner, using K-Means clustering to cluster frames using the image histogram and then select the best frame from clusters based on the variance of laplacian as the visual summary.

Table 3.4: Factual consistency evaluation results.

Datasets	MSMO	Img+Trans	TFN	HNNattTI	M ² SM	SCCS
CNN	40.12	41.23	41.52	42.33	42.59	44.37
DailyMail	50.31	50.65	50.72	51.37	51.69	53.16

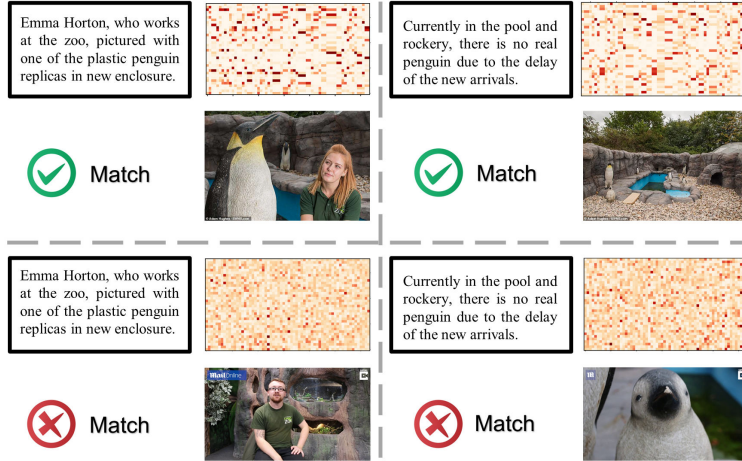


Figure 3.4: OT coupling: The OT coupling shows sparse patterns and specific temporal structure for the embedding vectors of ground-truth-matched video and text segments.

From Table 3.1 and Table 3.2, we can find that multimodal methods outperform unimodal approaches, showing the effectiveness of exploring the relationship and taking advantage of the cross-domain alignments of generating high-quality summaries.

Interpretation To show a deeper understanding of the multimodal alignment between the visual domain and language domain, we compute and visualize the transport plan to provide an interpretation of the latent representations, which is shown in Figure 3.4. When we regard the extracted embedding from both text and image spaces as the distribution over their corresponding spaces, we expect the optimal transport coupling to reveal the underlying similarity and structure. Also, the coupling seeks sparsity, which further helps to explain the correspondence between the text and image data.

Figure 3.4 shows comparison results of matched image-text pairs and non-matched ones. The top two pairs are shown as matched pairs, where there is an overlap between the image and the corresponding sentence. The bottom two pairs are shown as non-matched ones, where the overlapping of meaning between the image and text is relatively small. The correlation between the image domain and the language domain can be easily interpreted by the learned transport plan matrix. In specific, the optimal transport coupling shows the pattern of sequentially structured knowledge. However, for non-matched image-sentence pairs, the estimated couplings are relatively dense and barely contain any informative structure. As shown in Figure 3.4, we can find that the transport plan learned in the cross-domain alignment module demonstrates a way to align the features from different modalities to represent the key components. The visualization of the transport plan contributes to the interpretability of the proposed model, which brings a clear understanding of the alignment module.

Limitations Due to the absence of large evaluation databases, we only evaluated our method on three publicly available datasets that can be used for the MSMO task. The popular video databases, i.e., COIN and Howto100M datasets, can not be used in our task, since they lack narrations and key-step annotation. So a large evaluation database is highly needed for evaluating the performance of MSMO approaches. As the nature of the summarization task, human preference has an inevitable influence on the performance, since the ground-truth labels were provided by human annotators. It’s somehow difficult to quantitatively specify the quality of the summarization result, and current widely used evaluation metrics may not reflect the performance of the results very well. So we are seeking some new directions to find another idea for quality evaluation. The current setting is short videos & short documents, due to the constrain of available data. To extend the current MSMO to a more general setting, i.e., much longer videos or documents, new datasets should be collected. However, this requires huge human effort in annotating and organizing a high-value dataset, which is extremely time-consuming and labor-intensive. Nevertheless, we believe the MSMO task is promising and can provide valuable solutions to many real-world problems.

3.6 Conclusion

In this chapter, we proposed SCCS, a segment-level Semantics-Consistent Cross-domain Summarization model for the MSMO task. Our model decomposed the video & article into segments based on the content to preserve the structural semantics, and explored the cross-domain semantics relationship via optimal transport alignment at the segment level. The experimental results on three MSMO datasets show that SCCS outperforms previous summarization methods. We further provide interpretation by the OT coupling. Our approach provides a new direction for the MSMO task, which can be extended to many real-world applications.

Chapter 4

Unsupervised Multimodal Temporal Segmentation of Long Livestream Videos

In Chapter 3, we discuss the application of MSMO by learning cross-domain alignment, which builds on temporal segmentation to provide structural summaries. In addition, with online learning growing popular, Livestream videos have become a significant part of online learning, where design, digital marketing, creative painting, and other skills are taught by experienced experts in the sessions, making them valuable materials. However, Livestream tutorial videos are usually hours long, recorded, and uploaded to the Internet directly after the live sessions, making it hard for other people to catch up quickly. An outline will be a beneficial solution, which requires the video to be temporally segmented according to topics.

In this chapter, we discuss how to temporally segment another form of video, Livestream videos. We introduce a large Livestream video dataset named MultiLive, and formulate the temporal segmentation of the long Livestream videos (TSLLV) task. We propose LiveSeg, an unsupervised **L**ivestream video temporal **S**egmentation solution, which takes advantage of multimodal features from different domains. Our method achieves a 16.8% F1-score performance improvement compared with the state-of-the-art method. [347] has been patented by Adobe for production on Behance Livestream [197].

4.1 Introduction

Video temporal segmentation has become increasingly important since it is the basis for many real-world applications, i.e., video scene detection, shot boundary detection, etc. Video temporal segmentation can be considered an essential pre-processing step, and an accurate temporal segmentation result could benefit many other tasks. The video temporal segmentation methods lie in two directions: unimodal and multimodal approaches. Unimodal approaches only use the visual modality of the videos to learn scene change or transition in a supervised manner, while multimodal methods exploit available textual metadata and learn joint semantic representation in an unsupervised way.

A considerable amount of long Livestream videos are uploaded to the Internet every day, but it is challenging to understand the main content of the long video quickly. Traditionally, we can only have an inaccurate assumption by reading the video's title or using the control bar to manually

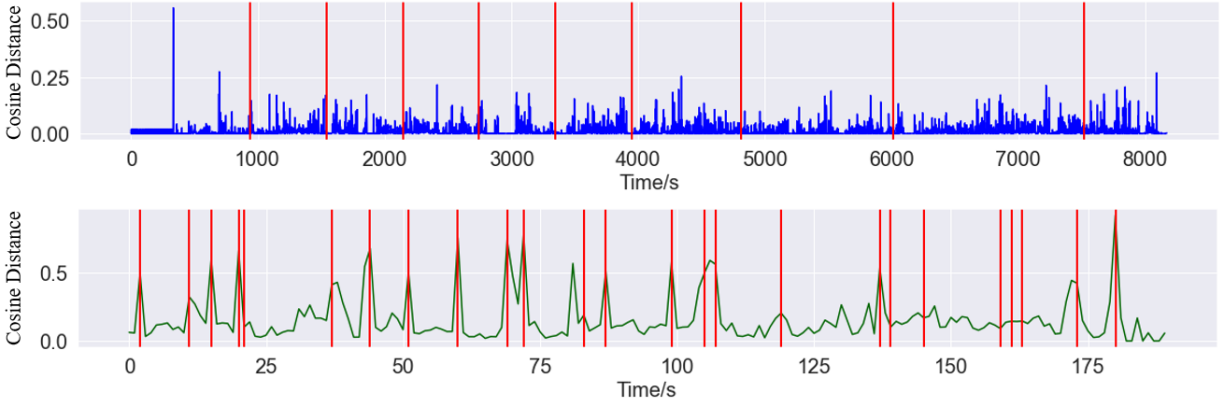


Figure 4.1: Comparison of temporal-pairwise cosine distance on visual features: (TOP) a Livestream video, (BOTTOM) a TVSum video (Blue & Green: distance; Red: segment boundaries).

access the video, which is time-consuming, inaccurate, and very easy to miss valuable information. An advantageous solution is to segment the long video into small segments based on the topics, making it easier for the users to navigate the content.

Most existing video temporal segmentation work focused on short videos. Some work explored movie clips extracted from long videos but easily segmented temporally by scene change. Jadon et al. proposed a summarization method based on the SumMe dataset [143], which are 1-6 min short videos with clear visual change [191]. When it comes to the long Livestream videos, the previous methods do not work well due to the extremely long length and new characteristics of the Livestream videos. So the critical problem is finding a practical approach to temporally segment the Livestream videos into segments. The quality of segmentation results can significantly impact further tasks. So here we propose a new task, TSLLV, temporal segmentation of long Livestream videos, which has not been explored yet. Different from other long videos, i.e., movies, Livestream videos usually contain more noisy visual information due to the visually abrupt change, and more noisy language information due to random chatting, conversational languages, and intermittent sentences, which means the content is neither clear nor well-organized, making it extremely hard to detect the segment boundaries. Comparison of the visual noisiness of the Livestream video and other videos and examples of Livestream transcripts are introduced in Section 4.3.

To sum up, the main difficulties for temporally segmenting the Livestream videos are:

- (1) The visual background remains similar for a considerable time, even though the topic has already changed, making the definition of boundaries ambiguous. For our MultiLive dataset collected from Behance¹, the hosts usually teach drawing or painting, where the main background is the board and remains similar for most parts of the video. Compared with movies, the movie’s background changes dramatically when switching to another scene, so the Livestream videos can not be split directly based on visual scene change or transition differences. Figure 4.1 shows an example comparison of temporal-pairwise cosine distance (distance between the i th frame and $i + 1$ th frame of the same video) on visual feature between a Livestream video and a TVSum video [431], which shows the Livestream video’s segment boundaries are not aligned with the visual scene change, making it difficult to segment.
- (2) The visual change is neither consistent nor clear. As shown in Figure 4.1, there are abrupt

¹<https://www.behance.net/live>

changes in the visual site due to the host changing folders or zooming in/out, making the visual information extremely noisy.

- (3) There is not enough labeled data for this kind of Livestream video, and it is challenging, time-consuming, and expensive to label them manually. Because it requires the human annotators to watch the entire video, understand the topics, and then temporally segment it, making it much more complicated than labeling images.

Our contributions are listed as follows:

- We introduce MultiLive, a new large dataset of Livestream videos, among which, 1,000 videos were manually segmented and annotated, providing human insights and references for evaluation.
- We formulate a new temporal segmentation of long Livestream videos (TSLLV) task according to the newly introduced MultiLive dataset.
- We proposed **LiveSeg**, an unsupervised **Livestream temporal Segmentation** method by exploring multimodal visual and language information as a solution to TSLLV. We extract features from both modalities, explore the relationship and dependencies across domains, and generate accurate segmentation results. LiveSeg achieved a 16.8% F1-score performance improvement compared with the SOTA method.

4.2 Related Work

Supervised Video Temporal Segmentation Temporal segmentation aims at generating small segments based on the content or topics of the video, which is easy to achieve when the video is short or when the scene change is easy to detect, e.g., in movie clips. Previous works mainly focused on short videos or videos with clear scene changes, which is convenient to manually label a huge amount of videos as training sets for supervised learning [4, 116, 217, 220, 327, 328, 342, 423, 428, 564].

Action, Shot, and Scene Segmentation Temporal action segmentation in videos has been widely explored [121, 224, 228, 403, 478, 489, 559]. However, those videos' characteristics are far different from Livestream videos, where the actions are well-defined, the main goal is to group similar actions based on visual change, and the length of videos is much shorter, so the methods can not be adopted directly. Shot boundary detection task is also very relevant and has been explored in many previous works [5, 152, 153, 453], where shot is defined by the visual change. However, in Livestream videos, segments are not solely defined by visual information, the topics contained in language also contribute to the definition of each segment. Video scene detection is the most relevant task. However, previous methods only used visual information to detect the scene change [56, 375, 390, 391, 548], so the methods can not be adopted directly for Livestream videos either.

Unsupervised Video Temporal Segmentation Recently, unsupervised methods have also been explored for video temporal segmentation. [202] proposed incorporating multiple feature sources with chunk and stride fusion to segment the video, but the datasets used are still short videos [143, 431]. [111] used Livestream videos as materials. However, they used internal software usage as the segmentation reference, which is not available for most videos, making their method highly restricted. Because for most videos, we can only get access to visual and audio/language metadata.

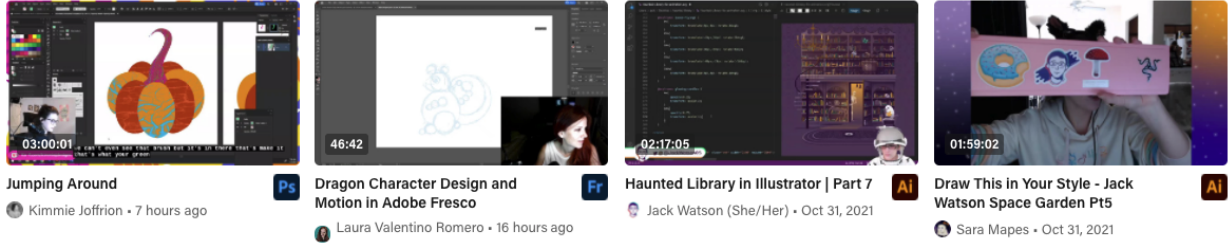


Figure 4.2: Example of Livestream videos.

Table 4.1: Distribution of video duration.

Video Duration	Number	Percentage
0-1 h	4,827	42.774%
1-2 h	2,945	26.097%
2-3 h	2,523	22.357%
3-4 h	705	6.247%
4-5 h	210	1.861%
5-6 h	70	0.620%
6-7 h	11	0.097%

Table 4.2: Distribution of transcript length.

Transcript Length	Number	Percentage
0-500	5,512	48.844%
500-1,000	2,299	20.372%
1,000-1,500	1,890	16.748%
1,500-2,000	989	8.746%
2,000-2,500	365	3.234%
2,500-3,000	118	1.046%
3,000-3,500	84	0.744%
3,500-4,000	35	0.310%
4,000-4,500	12	0.106%
4,500-5,000	3	0.027%

Summary Although previous models have shown reasonable results, they still suffer some drawbacks. Most work targeted short videos with clear scene changes instead of long videos and only used visual information while ignoring other domains, like language. Due to the characteristics of the Livestream videos in our MultiLive dataset, methods that solely depend on visual features can not obtain accurate results, so a multimodal approach should be addressed to incorporate visual and language information.

4.3 MultiLive Dataset

We introduce a large Livestream video dataset from Behance², which contains Livestream videos for showcasing and discovering creative work. The dataset includes video ID, title, video metadata, extracted transcript metadata from audio signals (by Microsoft ASR [512]), offset (timestamp), duration of each sentence, etc. The whole dataset contains 11,285 Livestream videos with a total duration of 15,038.4 hours, the average duration per video is 1.3 hours. The entire transcript contains 8,001,901 sentences, and the average transcript length for each video is 709 sentences. The detailed statistics of the dataset are shown in Table 4.1 and Table 4.2. From Tables 4.1,4.2, most videos are less than 3 hours, and most videos’ transcripts contain less than 1,500 sentences. In addition, we showed the histogram of video length distribution and transcript length distribution in Figure 4.3.

Besides, for the purpose of evaluation, we provide human annotations of 1,000 videos with segmentation boundaries annotated manually by human annotators for evaluation. The human annotators are asked to watch and understand the whole video and split each into several segments

²<https://www.behance.net/live>

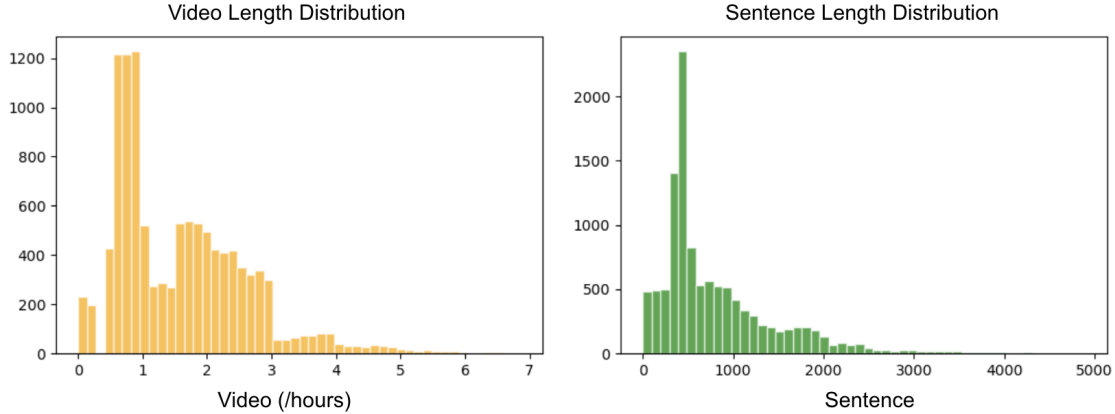


Figure 4.3: Histogram of MultiLive video length distribution and transcript length distribution (y-axis: number of videos).

Table 4.3: Comparison of MultiLive with existing datasets. SLR: scene length ratio.

Statistics	MultiLive	SumMe [143]	TVSum [431]	OVP [20]
Labeled videos	1,000	25	50	50
Ave. length (min)	78 mins	2.4 mins	4.2 mins	1.5 mins
Ave. scene num	8.8	5.1	52.2	8.8
Ave. SLR (min/scene)	8.86	0.47	0.08	0.17
Ave. SD	0.07	0.22	0.19	0.35

based on their understanding of the video content. The current 1,000 videos’ annotation includes 10 annotators from Amazon Mechanical Turk ³ (legal agreement signed). The annotators were separated into groups and each group watched part of the videos and then discussed the results together about the segmentation results to ensure the quality of the annotation was agreed upon by all the annotators. They were instructed to pay more attention to topic change, w.r.t. the moment that the live-streamer starts discussing a different topic.

There are several widely used video datasets in temporal segmentation or video summarization tasks [20, 143, 431], Table 4.3 shows the comparison of our dataset with the others. The amount of labeled videos of the others is less than 50, while we provide human annotations for 1,000 videos. The average length of the videos from our dataset is much longer than others, while the number of segments is in the same order of magnitude or even smaller than the others. The effect is that the average SLR (scene length ratio) of the Livestream dataset is much larger, where average SLR (scene length ratio) can be considered a metric to represent the average length of each scene in the video, calculated by (ave. length/ave. scene num). So the larger the ratio, the more content contained in each segment, leading to more difficulty finding the segment boundaries.

To demonstrate a more precise understanding of the visual information of Livestream videos, we compared the visual features extracted from one example Livestream video and one example TVSum video [431]. We extracted video frames from the raw video sequence, used ResNet50 model [156] (pre-trained on ImageNet) to extract the visual features of each video frame, and adopted t-SNE [466] to visualize the visual features. Figure 4.4(a) shows the Livestream video’s visual feature distribution, different colors with the same marker “o” representing different segments, ten

³<https://www.mturk.com/>

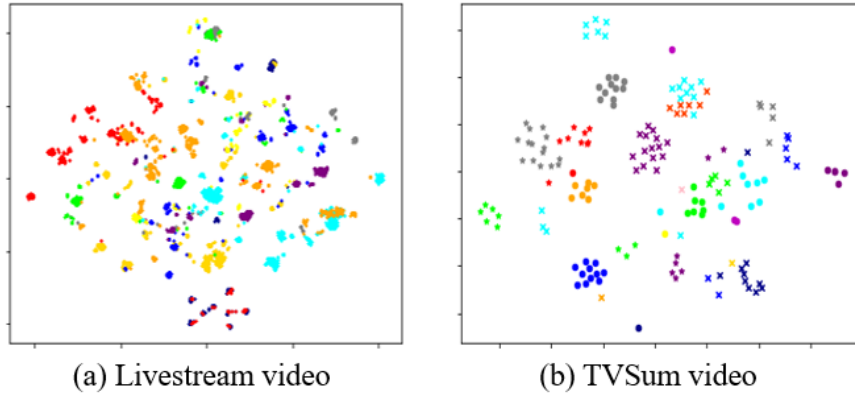


Figure 4.4: (a) Visual features of a Livestream video; (b) Visual features of a TVSum video, where different colors represent different segments within one video.

Table 4.4: Comparison of different types of videos.

Statistics	ASR WER	USR
Film Corpus [473, 474]	0.01	0.126
Movie Dialog Corpus [2]	0.01	0.139
MultiLive	0.05	0.458

segments in total. We can find that feature points that belong to different segments mix together and thus are hard to separate. As for TVSum’s video result in Figure 4.4(b), different colors or different markers “o”/“x”/“*” all represent different segments, 23 segments in total, which shows the points belong to different segments can be distinguished more easily than the Livestream video. This proves our statement that Livestream videos contain more noisy visual information, making it much harder to be temporally segmented by traditional methods.

Table 4.4 also shows the comparison of our Livestream data with movie datasets [2, 473, 474], which were collected from IMDB and TMDb, to emphasize the differences between Livestream videos and movies. Table 4.4 shows the Livestream videos’ ASR WER (word error rate) is higher than movies, and the USR (unrelated sentence rate) is much higher than movies, which contain more meaningless conversational languages. We further used hierarchical clustering to group the frames based on visual features and generated a dendrogram. As shown in Figure 4.5, we could find that the video frames far away from each other in timestamp can still be clustered together into the same group if only visual features are used. It supports the claim that using only visual information is insufficient to generate accurate temporal segmentation results, as the visual domain lacks sufficient information. So other domain features should be explored to provide more information.

To show a representative comparison, we computed the frame-level average distance (ave. SD) between the segments of our MultiLive dataset and the SumMe, TVSum, and OVP datasets. The results are shown in Table 4.3. The distance is computed on the two adjacent frames on each video segment boundary (last frame of i th segment, and first frame of $(i + 1)$ th segment, and the average results could show the average visual difference comparison. As in Table 4.3, we can find that the ave. SD of the MultiLive dataset is much smaller than the ave. SD of other datasets, which could be a representative metric to demonstrate that Livestream video’s visual change is much more noisy than existing datasets, making it more difficult to segment.

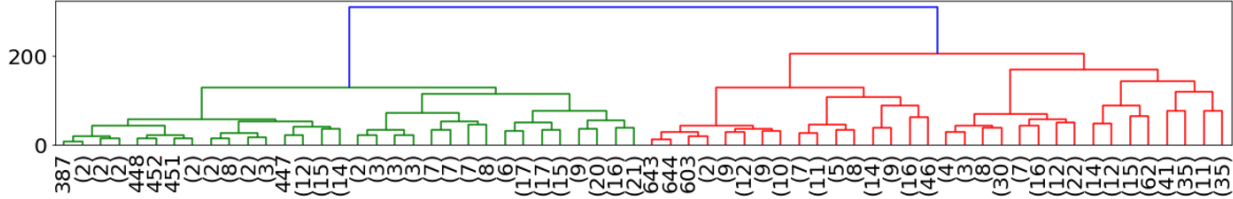


Figure 4.5: Dendrogram result of one Livestream video by hierarchical clustering of visual features, where the numbers below the bottom layer represent the number of images belonging to the corresponding sub-tree.

Table 4.5: Example of livestream transcripts.

Sentence	Offset	Transcript
1	79	Good morning, good morning. My name is Kara Sykes and I am in artist here.
2	91	My light is very bright this morning.
3	94	Sometimes you can turn it down.
12	137	I got more sleep than we have been getting so I was like I'm going Live Today.
20	166	Let's open up Photoshop Screen, but it's going to be we're gonna be working in illustrator.
31	204	Let's go ahead and create.
32	208	I've got a sketch, but I'm actually going to work just without it, but what I want to do here is create some lines.
112	568	Doing letters you never do letters, and I say, I know, but I really wanted to let her his name, so that's what I'm doing currently.
146	698	Now when I work for the area that I want to create, but let's just let's do this OK, I have my do not disturb on because at night we keep it off just so that doesn't wake up.
160	825	Tell you what these type people who create custom type you are amazing.

4.4 Proposed Method

The TSELLV task (temporal segmentation of long Livestream video) aims to accurately and temporally segment the Livestream videos based on the topics. Due to the absence of segmented labels and the time-consuming of manually labeling a huge amount of such long videos, we adopt unsupervised methods to segment the Livestream videos temporally. The whole framework is shown in Figure 4.6. Given a Livestream video \mathcal{S} , our target is to temporally segment video \mathcal{S} into $[S_1, S_2, \dots, S_k]$ based on topics, where k is the number of segments. The only available materials are video (visual input) and transcripts (language input). The number of segments of each to-be-segmented video is not preliminary given.

LiveSeg Framework The LiveSeg model takes input from the visual domain and the language domain. For visual features, we sample video frames $[f_1, f_2, \dots, f_n]$, where n is the timestamp, from the raw video \mathcal{S} (one frame per second to reduce the computation complexity). Then we use ResNet-50 [156] pre-trained on ImageNet [395] to extract visual features (fingerprints) $V_1 = [V_{11}, V_{12}, \dots, V_{1n}]$, where the visual fingerprints represent the video content. For the language features, due to the fact that the transcript is not temporally perfectly aligned with video frames, we first assign the transcript sentences to the corresponding video frame. If there are overlaps between several sentences or several frames, we duplicate those in a corresponding manner, and formulate

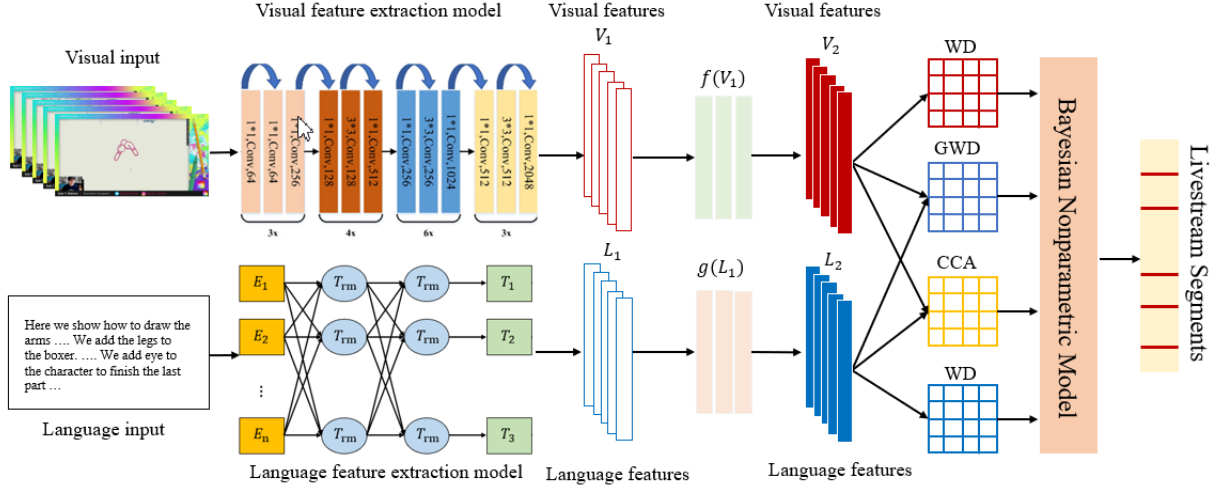


Figure 4.6: LiveSeg framework: The framework of LiveSeg for unsupervised multimodal Livestream video temporal segmentation.

frame-transcript pairs for each sampled frame in the timeline. Since the frames are sampled by a one-second time window, the transcripts are also aligned with each time window. If one transcript sentence T_i does not end for the given window, meaning the language has overlapped with two adjunct time windows, then we will assign this sentence T_i to both time window t and time window $t + 1$. We then extract sentence embeddings with BERT [86] to get sentence-level representations $L_1 = [L_{11}, L_{12}, \dots, L_{1n}]$. The embedding model used in our formulation is “all-MiniLM-L6-v2” from Sentence-Transformers [382], which is trained on large sentence level datasets using a self-supervised contrastive learning objective from pre-trained model [484] and fine-tuned on a sentence pairs dataset. Due to the ambiguity of the transcript, i.e., the examples shown in Table 4.5, redundant and noisy words are removed before generating language embeddings (redundant and noisy words mean the words that appear more than three times in a row due to the live-streamers speaking error).

The previous work [54, 194, 260], which took advantage of the alignment of vision and language features, inspired us to assume that there should be a relationship and dependency between visual and language features. [55, 62, 212, 256, 476] find that Optimal Transport shows tremendous power in sequence-to-sequence learning. In addition, [54, 540] find that Gromov Wasserstein Distance shows even better performance in measuring the distances in counterpart domains. Canonical Correlation Analysis, a well-known approach to exploring the correlation between different modalities, has been studied in many previous works for its ability to recognize the cross-domain relationship [16, 138, 517, 520]. [305, 456] showed that Bayesian Nonparametric Models performed well on temporal segmentation task, especially under unsupervised settings, which stands as a good candidate for our TSLLV task. Therefore, we adopt Deep Canonical Correlation Analysis [16] to encode the dependency for a hierarchical feature transformation. The networks transform raw visual features V_1 to high-level visual features V_2 with the transformation $f(V_1)$, and transform raw language features L_1 to high-level language features L_2 with the transformation $g(L_1)$. Then we compute the **WD** on the high-level temporal visual features V_2 and language features L_2 . We also calculate the Gromov Wasserstein Distance (**GWD**) and **CCA** on the two different modalities at the same timestamp, then use Bayesian Nonparametric models [200] to segment the Livestream videos temporally. The details of each part are introduced in the following paragraphs and sections.

Wasserstein Distance **WD** is introduced in **OT**, as in Chapter 2, which is a natural type of divergence for registration problems as it accounts for the underlying geometry of the space, and has been used for multimodal data matching and alignment tasks [54, 81, 146, 230, 361, 540]. In Euclidean settings, OT introduces WD $\mathcal{W}(\mu, \nu)$, which measures the minimum effort required to “displace” points across measures μ and ν , where μ and ν are values observed in the empirical distribution.

In our setting, we compute the temporal-pairwise Wasserstein Distance on both visual features and language features, considering each feature vector representing each frame or transcript embedding, which is $(\mu, \nu) = (V_{2i}, V_{2(i+1)})$ and $(\mu, \nu) = (L_{2j}, L_{2(j+1)})$ for $i, j \in t - 1$.

For simplicity without loss of generality, assume $\mu \in P(\mathbb{X})$ and $\nu \in P(\mathbb{Y})$ denote the two discrete distributions, formulated as $\mu = \sum_{i=1}^n u_i \delta_{x_i}$ and $\nu = \sum_{j=1}^m v_j \delta_{y_j}$, with δ_x as the Dirac function centered on x . $\Pi(\mu, \nu)$ denotes all the joint distributions $\gamma(x, y)$, with marginals $\mu(x)$ and $\nu(y)$. The weight vectors $u = \{u_i\}_{i=1}^n \in \Delta_n$ and $v = \{v_i\}_{i=1}^m \in \Delta_m$ belong to the n - and m -dimensional simplex, respectively. The WD between the two discrete distributions μ and ν is defined as:

$$\mathcal{WD}(\mu, \nu) = \inf_{\gamma \in \Pi(\mu, \nu)} \mathbb{E}_{(x, y) \sim \gamma} [c(x, y)] = \min_{\mathbf{T} \in \Pi(\mathbf{u}, \mathbf{v})} \sum_{i=1}^n \sum_{j=1}^m T_{ij} \cdot c(x_i, y_j) \quad (4.1)$$

where $\Pi(u, v) = \{T \in \mathbb{R}_+^{n \times m} \mid T \mathbf{1}_m = u, T^\top \mathbf{1}_n = v\}$, $\mathbf{1}_n$ denotes an n -dimensional all-one vector, and $c(x_i, y_j)$ is the cost function evaluating the distance between x_i and y_j . The temporal-pairwise WD on both visual and language features encodes the temporal difference and consistency within the same domain.

Gromov Wasserstein Distance Classic OT requires defining a cost function across domains, which can be challenging to implement when the domains are in different dimensions [380]. **GWD** [340] extends OT by comparing distances between samples rather than directly comparing the samples themselves. Assume there are metric measure spaces (\mathcal{X}, d_x, μ) and (\mathcal{Y}, d_y, ν) , where d_x and d_y are distances on \mathcal{X} and \mathcal{Y} , respectively. We compute pairwise distance matrices D^x and D^y as well as the tensor $L \in \mathbb{R}^{n_x \times n_x \times n_y \times n_y}$, where $L_{ijkl} = L(D_{ik}^x, D_{jl}^y)$ measures the distance between pairwise distances in the two domains. Intuitively, $L(d_x(x_1, x_2), d_y(y_1, y_2))$ captures how transporting x_1 onto y_1 and x_2 onto y_2 would distort the original distances between x_1 and x_2 and between y_1 and y_2 . The discrete Gromov-Wasserstein problem is then defined by:

$$\mathcal{GWD}(p, q) = \min_{\Gamma \in \Pi(p, q)} \sum_{i, j, k, l} L_{ijkl} \Gamma_{ij} \Gamma_{kl} \quad (4.2)$$

where $(p, q) = (V_{2k}, L_{2k})$ is the visual-language feature pairs. For each tuple (x_i, x_k, y_j, y_l) , we compute the cost of altering the pairwise distances between x_i and x_k when splitting their masses to y_j and y_l by weighting them by Γ_{ij} and Γ_{kl} , respectively. In our framework, the computed GWD across domains captures the relationships and dependencies between visual and language domains.

CCA and DCCA **CCA** is a method for exploring the relationships between two multivariate sets of variables, which can learn the linear transformation of two vectors in order to maximize the correlation between them, which is used in many multimodal problems [16, 28, 129, 266, 267, 352]. In our problem, we apply CCA to capture the cross-domain relationship of visual features V_{2l} and

language features L_{2l} . For visual features V_{2l} and language features L_{2l} , where $l \in t$. We assume $(V_{2l}, L_{2l}) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2}$ has covariances $(\Sigma_{11}, \Sigma_{22})$ and cross-covariance Σ_{12} . CCA finds pairs of linear projections of the two views, $(w'_1 V_{2l}, w'_2 L_{2l})$ that are maximally correlated:

$$(w_1^*, w_2^*) = \operatorname{argmax}_{w_1, w_2} \operatorname{corr}(w'_1 V_{2l}, w'_2 L_{2l}) = \operatorname{argmax}_{w_1, w_2} \frac{w'_1 \Sigma_{12} w_2}{\sqrt{w'_1 \Sigma_{11} w_1 w'_2 \Sigma_{22} w_2}} \quad (4.3)$$

Since the objective is invariant to scaling of w_1 and w_2 , the projections are constrained to have unit variance:

$$(w_1^*, w_2^*) = \operatorname{argmax}_{w'_1 \Sigma_{11} w_1 = w'_2 \Sigma_{22} w_2 = 1} w'_1 \Sigma_{12} w_2 \quad (4.4)$$

To obtain V_2 and L_2 , DCCA is applied in the framework for nonlinear feature transformation. If we assign θ_1 and θ_2 to represent the parameters for $f(V_1)$ and $g(L_1)$, respectively, where V_1 and L_1 represent the low-level visual and language features, then the transformation aims at:

$$(\theta_1^*, \theta_2^*) = \operatorname{argmax}_{(\theta_1, \theta_2)} \operatorname{corr}(f(V_1; \theta_1), g(L_1; \theta_2)) \quad (4.5)$$

The parameters are trained to optimize this quantity using gradient-based optimization by taking the correlation as the negative loss with backpropagation to update the nonlinear transformation model [16].

Bayesian Nonparametric Model We used Hierarchical Dirichlet Process Hidden semi-Markov Model (HDP-HSMM) to generate the video segments for modeling [113, 200], which can infer arbitrarily large state complexity from sequential and time-series data.

The process of HDP-HSMM is illustrated in Figure 4.7. In the model, z_i denotes the classes of the segments, β denotes an infinite-dimensional multinomial distribution, which is generated from the GEM distribution and parameterized by γ [341]. GEM denotes the co-authors Griffiths, Engen, and McCloskey, with the so-called stick-breaking process (SBP) [415]. The probability π_i denotes the transition probability, which is generated by the Dirichlet process and parameterized by β [457]:

$$\beta \mid \gamma \sim \text{GEM}(\gamma), \quad \pi_i \mid \alpha, \beta \sim \text{DP}(\alpha, \beta), \quad i = 1, 2, \dots, \infty \quad (4.6)$$

where γ and α are the concentration parameters of the Dirichlet processes (DP). The probability distribution is constructed through a two-phase DP named Hierarchical Dirichlet process (HDP) [305, 457]. The class z_i of the i th segment is determined by the class of the $(i - 1)$ th segment and transition probability π_i .

In HSMM, state transition probability from state i to j can be defined as $\pi_{i,j} = p(x_{t+1} = j \mid x_t = i)$, then the transition matrix can be denoted as $\pi = \{\pi_{i,j}\}_{i,j=1}^{|\mathcal{X}|}$, where $|\mathcal{X}|$ denotes the number of hidden states. The distribution of observations y_t given specific hidden states is denoted by $p(y_t \mid x_t, \theta_i)$, where θ_i denotes the emission parameter of state i . Then the HSMM can be described as:

$$x_s \mid x_{s-1} \sim \pi_{x_{s-1}}, \quad d_s \sim g(\omega_s), \quad y_t \mid x_s, d_s \sim F(\theta_{x_s}, d_s) \quad (4.7)$$

where $F(\cdot)$ is an indexed family of distributions, the probability mass function of d_s is $p(d_t \mid x_t = i)$, $g(\omega_s)$ denotes a state-specific distribution over the duration d_s , and ω_s denotes the parameter priori of the duration distributions.

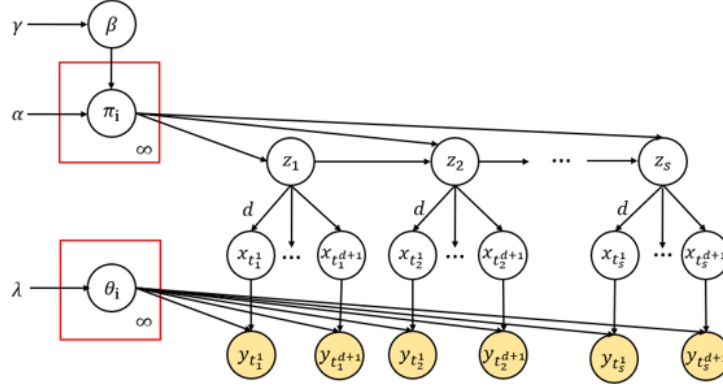


Figure 4.7: Graphical model of HDP-HSMM

In HDP, let Θ be a measurable space with a probability measure H on the space, γ is a positive real number named the concentration parameter. $\text{DP}(\gamma, H)$ is defined as the distribution of the random probability measure of G over Θ . For any finite measurable partition of Θ , the vector is distributed as a finite-dimensional Dirichlet distribution:

$$G_0 \sim \text{DP}(\gamma, H), G_0 = \sum_{k=1}^K \beta_k \delta_{\theta_k}, \theta_k \sim H, \beta \sim \text{GEM}(\gamma) \quad (4.8)$$

where θ_k is the distribution of H , $\beta \sim \text{GEM}(\gamma)$ represents the stick-breaking construction process of the weight coefficient [326, 565], and δ_θ is the Dirac function. The model can be written as:

$$\theta_i \sim H(\lambda), i = 1, 2, \dots, \infty, z_s \sim \bar{\pi}_{z_{s-1}}, s = 1, 2, \dots, S \quad (4.9)$$

$$D_s \sim g(\omega_{z_s}), s = 1, 2, \dots, S, \omega_i \sim \Omega \quad (4.10)$$

$$x_{t_1^s:t_s^{D_s+1}} = z_s, y_{t_1^s:t_s^{D_s+1}} \sim F(\theta_{x_t}) \quad (4.11)$$

where π_i is the distribution parameter of hidden state sequence z_s , implying that HDP provides an infinite number of states for HSMM, D_s is the length distribution of the state sequence with distribution parameter ω , and y_{t_s} is the observation sequence with distribution parameter θ_i [346].

For parameter inference of the HDP-HSMM model, a weak-limit Gibbs sampling algorithm is applied [200]. The weak limit approximation transforms the infinite dimension hidden state into finite dimension form so that the hidden state chain can be updated according to the observation data [346]. It is assumed that the basic distribution $H(\cdot)$ and the observation series distribution $F(\cdot)$ are conjugated distributions, the hidden states distribution $g(\cdot)$ is a Poisson distribution, and the hidden states distribution and the observation series distribution are independent. We first sample the weight coefficient β and the state sequence distribution parameter π_i :

$$\beta \mid \gamma \sim \text{Dir}(\gamma/S, \dots, \gamma/S), \pi_i \mid \alpha, \beta \sim \text{Dir}(\alpha\beta_1, \dots, \alpha\beta_s) j = 1, \dots, S \quad (4.12)$$

Then we sample the observation distribution parameters θ_i and state duration distribution parameter ω_i according to observation data. It is assumed that the observed data obey a multivariate Gaussian distribution, the model parameters $\theta_i = (u_i, \Sigma_i)$ obey the Normal–Inverse–Wishart distribution:

$$\text{NIW}(u, \Sigma \mid v_0, \Delta_0, \mu_0, S_0) \triangleq \text{N}(\mu \mid \mu_0, S_0) * \text{IW}(\Sigma \mid v_0, \Delta_0) \quad (4.13)$$

where $\varphi = \{u_0, S_0, v_0, \Delta_0\}$ are prior parameters, μ_0 and S_0 are the prior mean and co-variance matrices, and ν_0 and Δ_0 are the degrees of freedom and scale of NIW distribution. The state duration distribution is a Poisson distribution, and parameter ω_i follows a Beta distribution: $\omega \sim \text{Beta}(\eta_0, \sigma_0)$. Then we update parameters according to the observation data [110, 200].

Final Segmentation Boundaries The raw output from the Bayesian nonparametric model contains both short and long segments, but the short segment may not contain comprehensive information, which will be useless as the final results. So we used a heuristic-based method to group the small segments into the large ones. The method is straightforward, where a parameter l_s defines the minimum length of the generated segments, if there is a segment shorter than l_s , then we compute the visual and textual similarity of this small segment with the two adjacent segments, and group the small segment into the one with higher similarity. Since these small segments are mostly due to the live-streamer abruptly zooming in/out or randomly chatting about something unrelated to the main topic, which has little influence on the segmentation results (i.e., a small part inside a big chunk), we just used this simple method to make the results look cleaner. The method introduced a parameter l_s , defined as the minimum length of the generated segments, which is used to hierarchically group the generated small segments into the bigger ones to eliminate the effect caused by small segments.

4.5 Experiments and Results

Baselines We select several representative baseline methods for comparison, which include:

- **Hierarchical Cluster Analysis (HCA)** HCA aims at finding discrete groups with varying degrees of similarity represented by a similarity matrix [70, 101], which produces a dendrogram as the intermediate result. The distance for the Livestream video setting is defined as: $d = \alpha_b d_t + (1 - \alpha_b) d_f$, where d_t is the timestamp distance, d_f is the feature content distance, and α_b is a balance parameter. Feature points representing content get separated further apart when the time distance of corresponding features is large.
- **TransNet V2** Soucek et al. proposed the TransNet V2 model for shot transition detection [433], which can also generate segmentation results and showed better performance than the previous method [453].
- **Hecate** Song et al. proposed the Hecate model to generate thumbnails, animated GIFs, and video summaries from videos [430], where shot boundary detection is one of the steps. This step will be used to compare with the other baseline methods as well as our method.
- **Optimal Sequential Grouping (OSG)** Rotman et al. proposed video scene detection algorithms based on the optimal sequential grouping [390, 391], which included finding pairwise distances between feature vectors and splitting shots into non-intersecting groups by optimizing a distance-based cost function.
- **LGSS** [375] proposed a local-to-global scene segmentation framework (LGSS), which used multiple features extracted by ResNet50, Faster-RCNN [384], TSN [478], and NaverNet [69]. The temporal segmentation step is based on PySceneDetect [47].

Temporal Segmentation on MultiLive Dataset For Livestream videos, the raw visual feature dimension is 2,048, and the raw language feature dimension is 384 extracted by pre-trained-

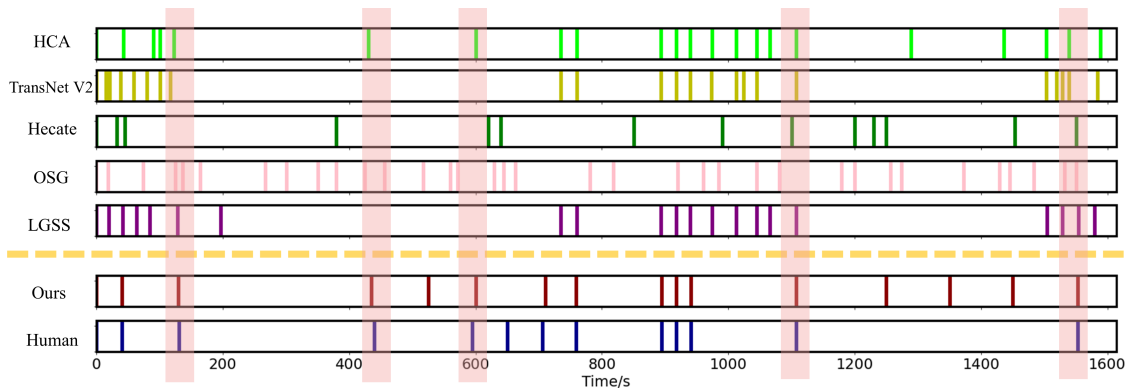


Figure 4.8: Comparison of boundary candidates by different methods, from top to bottom: (1) HCA, (2) TransNet V2, (3) Hecate, (4) OSG, (5) LGSS, (6) ours (LiveSeg), and (7) Human Annotations.

Table 4.6: Comparison of segmentation results.

Methods	Backbone	Modality	Precision	Recall	F1-score
HCA [70]	HCA	Visual	0.482	0.487	0.484
TransNet V2 [453]	ResNet-18	Visual	0.536	0.525	0.530
Hecate [430]	Clustering	Visual	0.539	0.533	0.536
OSG [391]	DP	Visual	0.574	0.557	0.565
LGSS [375]	Bi-LSTM	Visual	0.587	0.581	0.584
LiveSeg-Visual	LiveSeg	Visual	0.591	0.666	0.626
LiveSeg-Language	LiveSeg	Language	0.589	0.568	0.578
LiveSeg-Multimodal	LiveSeg	Multimodal	0.673	0.697	0.685

BERT models from Huggingface⁴. For the hierarchical transformation performed by DCCA. In our experiments, we set l_s to one minute. To make a fair comparison, we also applied the post-processing step in Sec 4.4 to all the baselines.

Due to the characteristics of Livestream videos, the video frames and transcripts are not perfectly aligned with the segments. Besides, different people segment the same video differently due to human preferences, which also needs to be considered. We performed the TSLLV task using baseline methods in Section 4.5 and our LiveSeg method on the MultiLive dataset. We evaluate the performance of different methods on the 1,000 annotated videos. Comparison results of baseline methods, our method, and human annotations for one Livestream video are shown in Figure 4.8. We can see that scene transition detection methods will generate inaccurate segments as the visual change is noisy for Livestream videos. However, many essential boundaries will be missed if simply improving the clustering threshold. Compared with existing methods, our results are more accurate and can be comparable with human annotations.

For quantitative analysis, tolerance interval ω_t is introduced. The correctness of the segmentation is judged at each position of this interval: a false alarm is declared if the algorithm claims a boundary in the interval while no reference boundary exists in the interval, and a miss is declared if the algorithm does not claim a boundary in the interval while a reference boundary exists in the interval [104]. In our experiment, we set ω_t to one minute, and we adopt precision, recall, and F1-score metrics to compare the performance of our results with human annotations. As shown in Table 4.6, our segmentation results outperform other baseline results. Besides, considering modality,

⁴<https://huggingface.co/>

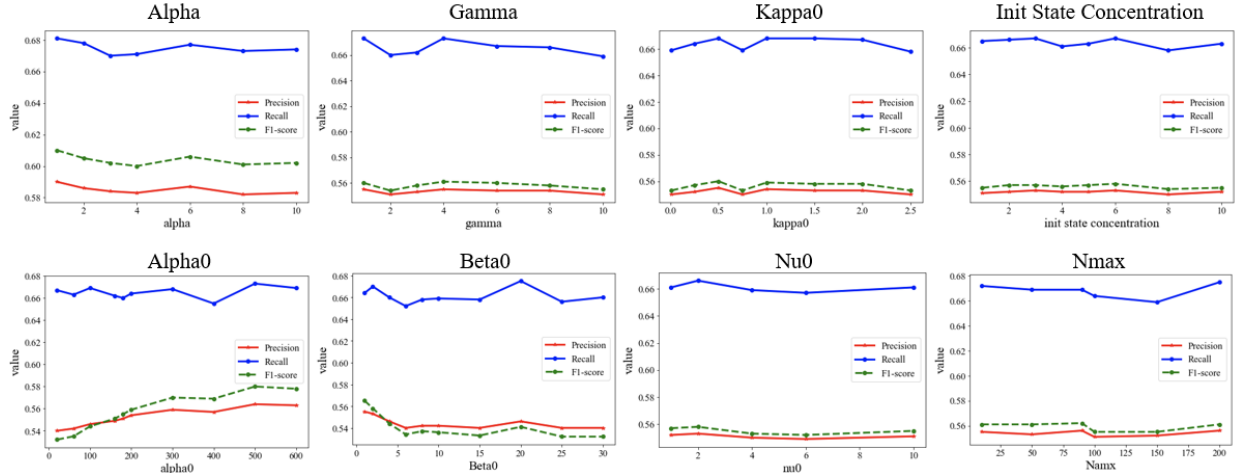


Figure 4.9: Segmentation performance with different parameters (Red: precision; Blue: recall; Green: F1-score).

Table 4.7: Comparison of performance with different interval ω_t .

ω_t	Precision	Recall	F1-score
0.5 min	0.608	0.672	0.627
1.0 min	0.673	0.697	0.685
1.5 min	0.605	0.666	0.621
2.0 min	0.600	0.659	0.615
2.5 min	0.598	0.653	0.610
3.0 min	0.595	0.647	0.605

multimodal segmentation outperforms single modality results, showing that the relationship learned between the visual domain and the language domain can truly benefit temporal segmentation.

Ablation Study Multiple heuristics could have an impact on the segmentation performance, such as tolerance interval ω_t and the parameters in the Bayesian nonparametric model. We carried out several ablation experiments on the influence of different parameters with multimodal features, where the results are shown in Table 4.7 and Figure 4.9.

We also provided an ablation study on different components, since only using WD is the same as LiveSeg-Visual and LiveSeg-Language, so we provide additional ablation study results on GWD and CCA. In Table 4.8, we can find that combining all of them (LiveSeg-Multimodal) can achieve better performance than using only one of the components.

Comparison on Other Datasets In addition, we compare our method with state-of-the-art unsupervised video summarization method [17] on the famous video summarization benchmark datasets, SumMe [143] and TVSum [431]. We used the same key-fragment-based approach for evaluation [551], where the similarity between a machine-generated and a user-defined ground-truth summary is represented by expressing their overlap using the F-Score. For a given video and a machine-generated summary, this protocol matches the latter against all the available user summaries for this video and computes a set of F-Scores. Table 4.9 shows the comparison F1-score of our method with SUM-GAN [17], our method can still show slightly better results on the SumMe

Table 4.8: Ablation study of different components.

	Precision	Recall	F1-score
GWD	0.622	0.673	0.646
CCA	0.603	0.654	0.615
WD (LiveSeg-Visual)	0.591	0.666	0.605
WD (LiveSeg-Language)	0.589	0.568	0.606

Table 4.9: Comparison with SOTA unsupervised baseline on traditional video summarization datasets.

F1-score	SUM-GAN [17]	LiveSeg
SumMe	50.8	51.3
TVSum	60.6	60.9

dataset and competitive results on the TVSum dataset, which clearly demonstrated the effectiveness of our method.

Limitation and Future Work The current method targets long Livestream videos, which shows better performance than existing ones, given that the current setting of both visual input and language input are highly noisy. However, it may not work better than supervised methods on short videos where scene change is clear, under which supervised methods could perform better when large-scale labeled training samples are available. However, the collected training samples highly constrain the generability and robustness of those approaches.

Due to the fact that labeling large-scale, long videos is time-consuming and expensive, so the current annotation result can be considered as the average result, which ensures the general quality, while may not preserve the nature that individual annotators may have different preferences, which can be useful as user-study materials. In future work, we will execute annotations for the same videos by different annotators separately, for evaluation and verification, which could provide human upper bound and future insights.

4.6 Conclusion

In this chapter, we propose LiveSeg, an unsupervised multimodal framework, focusing on the temporal segmentation of long Livestream video (TSLLV) task, which has not been explored before. We collect a large Livestream video dataset named MultiLive, and provided human annotations of 1,000 Livestream videos for evaluation. By quantitative analysis and human evaluation of our experimental results, we demonstrate that our model is able to generate high-quality temporal segments, which establishes the basis for Livestream video understanding tasks and can be extended to many real-world applications.

Chapter 5

MMSum: A Dataset for Multimodal Summarization

In Chapters 3,4, we discuss that **MSMO** has emerged as a promising research direction. Nonetheless, numerous limitations exist within existing public **MSMO** datasets, including insufficient maintenance, data inaccessibility, limited size, and the absence of proper categorization, which pose significant challenges. To address these challenges and provide a comprehensive dataset for this new direction, we collect a new dataset named **MMSum**. Our new dataset features (1) Human-validated summaries for both video and textual content, providing superior human instruction and labels for multimodal learning. (2) Comprehensively and meticulously arranged categorization, spanning 17 principal categories and 170 subcategories to encapsulate a diverse array of real-world scenarios. (3) Benchmark tests performed on the proposed dataset to assess various tasks and methods, including *video summarization*, *text summarization*, and *multimodal summarization*. To champion accessibility and collaboration, we release the **MMSum** dataset and the data collection tool as fully open-source resources, fostering transparency and accelerating future developments. Our project website can be found at <https://mmsum-dataset.github.io/>, which includes our dataset and codebase.

5.1 Introduction

MSMO is an emerging research topic spurred by advancements in multimodal learning [52, 172, 210, 309, 576] and the increasing demand for real-world applications such as medical reporting [271], educational materials [371], and social behavior analysis [283]. Most **MSMO** studies focus on video data and text data, aiming to select the most informative visual keyframes and condense the text content into key points. In this chapter, we continue working on **MSMO**, which integrates both visual and textual information to provide users with comprehensive and representative summaries to enhance user experience [114, 245, 576].

Despite the respective accomplishments of conventional unimodal summarization techniques on video data [195, 332, 388, 551, 558, 566, 578] and text data [63, 293, 307, 308, 561], multimodal summarization continues to pose challenges due to a number of complexities. (1) The intricate nature of multimodal learning necessitates an algorithm capable of exploiting correlated information across different modalities, (2) There is a scarcity of appropriate multimodal datasets that reliably

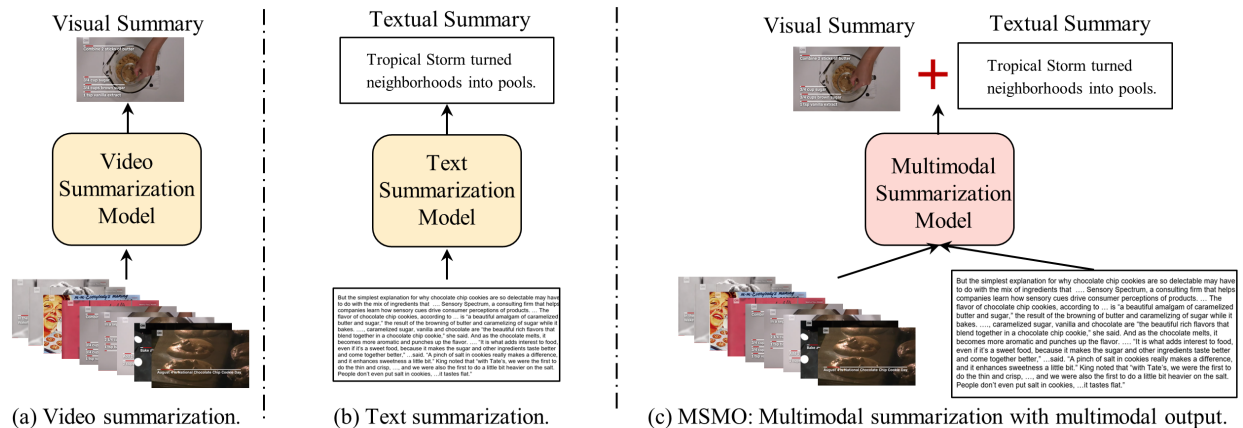


Figure 5.1: Task comparison: (a) traditional video summarization, (b) text summarization, and (c) MSMO tasks.

exhibit cross-modal correlations across diverse categories, and (3) There exists a gap in comprehensive evaluation protocols that accurately reflect the efficacy of MSMO methods in terms of their performance on both intermediate interpretations and downstream tasks.

Merging existing video and text datasets appears to be a feasible approach. However, assuring the presence of cross-modal correlations proves challenging [309], not to mention the absence of necessary human verification [325], a vital element in machine learning research. Furthermore, the existing datasets pose several issues, such as inadequate maintenance leading to data unavailability, limited size, and lack of categorization. To address these concerns and offer a comprehensive dataset for this area of study, we have undertaken the task of collecting a new dataset, named **MMSum**. Our contributions are summarised as follows:

- **A new MSMO dataset** Introducing MMSum, our newly curated MSMO dataset, specifically designed to cater to a wide range of tasks, with a particular emphasis on MSMO. This extensive dataset offers abundant information that serves as solid support for various research topics.
- **Diverse categorization** Within the MMSum dataset, we have gathered videos spanning 17 primary categories. Each of these main categories further comprises 10 distinct subcategories, culminating in a grand total of 170 subcategories. This comprehensive categorization ensures that the MMSum dataset is exceptionally representative and encompasses a wide range of content.
- **New benchmark** Across a diverse array of tasks, our results can be regarded as a benchmark on this novel real-world dataset.
- **Accessibility** We open-source the MMSum dataset and the corresponding data collection tool with CC BY-NC-SA License.

5.2 Related Work

Unimodal Summarization typically comprises video summarization and text summarization. Video summarization involves extracting key moments that summarize the content of a video by selecting the most informative and essential parts. Traditional video summarization methods

primarily rely on visual information. However, recent advancements have introduced category-driven or supervised approaches that generate video summaries by incorporating video-level labels, thereby enhancing the summarization process [149, 310, 431, 506, 566, 567]. Text Summarization involves processing textual metadata, such as documents, articles, tweets, and more, as input, and generating concise textual summaries. The quality of generated summaries has recently been significantly improved through fine-tuning pre-trained language models [268, 555].

Video Summarization Video summarization aims to generate a short synopsis that summarizes the video content by selecting the most informative and vital parts. The summary usually contains a set of representative video keyframes or video key fragments that have been stitched in chronological order to form a shorter video. The former type is known as video storyboard, and the latter one is known as video skim [18]. Traditional video summarization methods only use visual information, extracting important frames to represent the video content. For instance, [143, 191] generated video summaries by selecting keyframes using SumMe and TVSum datasets. Some category-driven or supervised training approaches were proposed to generate video summaries with video-level labels [431, 506, 566, 567].

Textual Summarization Textual summarization takes textual metadata, i.e., documents, articles, tweets, etc, as input and generates textual summaries in two directions: abstractive summarization and extractive summarization. Abstractive methods select words based on semantic understanding, and even the words may not appear in the source [411, 450]. Extractive methods attempt to summarize language by selecting a subset of words that retain the most critical points, which weights the essential part of sentences to form the summary [312, 505]. Recently, the fine-tuning approaches have improved the quality of generated summaries based on pre-trained language models in a wide range of tasks [268, 555].

Video Temporal Segmentation Video temporal segmentation aims at generating small video segments based on the content or topics of the video, which is a fundamental step in content-based video analysis and plays a crucial role in video analysis. Previous work mostly formed a classification problem to detect the segment boundaries in a supervised manner [4, 342, 423, 428, 564]. Recently, unsupervised methods have also been explored [143, 431]. Temporal segmentation of actions in videos has also been widely explored in previous works [224, 228, 403, 478, 489, 559]. Video shot boundary detection and scene detection tasks are also relevant and has been explored in many previous studies [56, 152, 153, 375, 548], which aim at finding the visual change or scene boundaries.

Textual Segmentation Textual segmentation aim at dividing the text into coherent, contiguous, and semantically meaningful segments [318]. These segments can be composed of words, sentences, or topics, where the types of text include blogs, articles, news, video transcripts, etc. Previous work focused on heuristics-based methods [66, 222], LDA-based modeling algorithms [39, 51], or Bayesian methods [51, 386]. Recent developments in NLP developed large models to learn huge amounts of data in a supervised manner [237, 292, 336, 487]. Besides, unsupervised or weakly-supervised methods has also drawn much attention [126, 275].

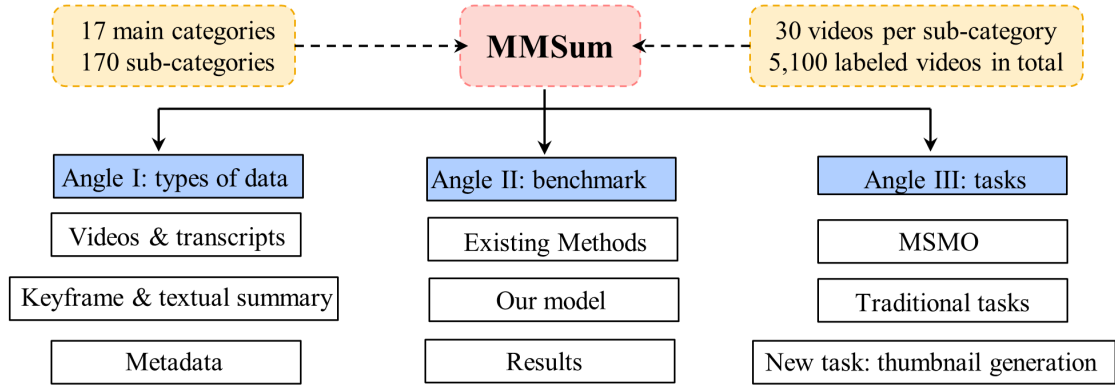


Figure 5.2: MMSum: The design of the proposed MMSum dataset is driven by research and application needs.

Multimodal Summarization explored multiple modalities for summary generation. [114, 324, 494, 541] learned the relevance or mapping in the latent space between different modalities. In addition to only generating visual summaries, [19, 236, 576] generated textual summaries by taking audio, transcripts, or documents as input along with videos or images, using seq2seq model [445] or attention mechanism [24]. The methods above explored using multiple modalities’ information to generate a single modality output, either textual or visual summary. Recent trends on the MSMO task have also drawn much attention [114, 115, 154, 295, 360, 361, 452, 552, 557, 576]. Specifically, [452] summarized a video and text document into a cover frame and a one-sentence summary. The most significant difference between multimodal summarization and MSMO lies in the inclusion of multiple modalities in the output.

5.3 Angle I: Types of data

5.3.1 Data Collection

In light of the aforementioned challenges inherent in the existing MSMO datasets, we propose a novel dataset named MMSum to address these issues comprehensively and effectively. Our approach involved the collection of a multimodal dataset, primarily sourced from a diverse range of untrimmed videos from YouTube. The collected dataset comprises a rich set of information, including video files and transcripts, accompanied by corresponding video metadata. Additionally, temporal boundaries were meticulously recorded for each segment within the videos. Furthermore, for each segment, we obtained both video summaries and text summaries. It is worth noting that these summaries were directly provided by the authors of the respective videos, ensuring their authenticity and reliability. Moreover, the dataset incorporates comprehensive video metadata, such as titles, authors, URLs, categories, subcategories, and so on. By gathering this diverse range of multimodal data and leveraging the ground-truth video and text summaries provided by the original content creators, we aim to create a valuable and reliable resource.

Fidelity Given the limited availability of fully annotated videos with complete and non-missing video summaries and text summaries, we resorted to a manual collection of videos that satisfied all the specified criteria. The meticulous nature of this process ensured that only videos meeting the



Figure 5.3: MMSum categories: The 17 main categories of the MMSum dataset, where each main category contains 10 subcategories, resulting in 170 subcategories in total.

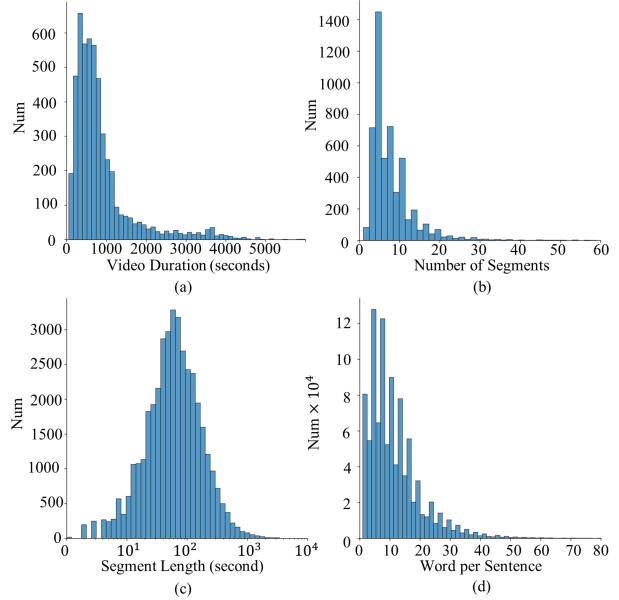


Figure 5.4: MMSum statistics: The statistics of the MMSum dataset, which show the distribution of (a) video duration; (b) number of segments per video; (c) segment duration; (d) number of words per sentence.

stringent requirements were included in the dataset. To illustrate the disparities between different tasks and datasets in terms of modalities, we provide a comprehensive comparison in Table 5.1. For instance, traditional video or text summarization datasets typically encompass either visual or textual information exclusively. While there are datasets available for traditional multimodal summarization, where multiple modalities are used as input, they still produce single-modality summaries. In contrast, the MSMO dataset holds significant value in real-world applications, as it requires multimodal inputs and provides summaries containing both visual and textual elements. Consequently, the collection process for this dataset necessitates acquiring all the requisite information, resulting in a time-consuming endeavor.

Human Verification Notably, every video in the MMSum dataset undergoes manual verification to ensure high-quality data that fulfills all the specified requirements. For the fidelity verification process, five human experts (3 male and 2 female) each spent 30 days watching the collected videos, understanding the content, and verifying the annotations. The annotators were instructed to pay specific attention to the quality of segmentation boundaries, visual keyframes, and textual summaries. The pre-filtered size of the dataset is 6,800 (40 videos per subcategory). After manual verification and filtering, only 30 of 40 are preserved to ensure the quality, resulting in the current size of 5,100 (30 videos per subcategory).

Diversity During the dataset creation process, we extensively examined existing video datasets such as [290, 569] for reference. Subsequently, we carefully selected 17 main categories to ensure comprehensive coverage of diverse topics. These main categories encompass a wide range of subjects, including *animals, education, health, travel, movies, cooking, job, electronics, art,*

Table 5.1: Comparison of the modality of different summarization tasks and datasets. Difference between traditional multimodal summarization and MSMO: traditional multimodal summarization still outputs a single-modality summary, while MSMO outputs both modalities’ summaries. Public Availability means whether the data is still publicly available and valid. Structural Summaries means available summaries of each segment, not just for the whole video.

Tasks	Datasets	Input		Output		Public Availability	Categorization	Structural Summaries
		Visual	Textual	Visual	Textual			
Video	TVSum [431]	✓	✗	✓	✗	✓	✗	✓
	SumMe [143]	✓	✗	✓	✗	✓	✗	✓
	VSUMM [80]	✓	✗	✓	✗	✓	✗	✓
Textual	X-Sum [311]	✗	✓	✗	✓	✓	✗	✗
	Pubmed [414]	✗	✓	✗	✓	✓	✗	✗
Multimodal	How2 [401]	✓	✓	✓	✗	✓	✗	✗
	AVIATE [19]	✓	✓	✗	✓	✓	✗	✗
	Daily Mail [576]	✓	✓	✗	✗	✓	✗	✗
MSMO	VMSMO [295]	✓	✓	✓	✓	✗	✗	✗
	MM-AVS [114]	✓	✓	✓	✓	✓	✗	✗
	MMSum (Ours)	✓	✓	✓	✓	✓	✓	✓

Table 5.2: Comparison with existing video summarization and multimodal summarization datasets.

	SumMe [143]	TVSum [431]	OVP [20]	CNN [115]	Daily Mail [115]	Ours
Source	YouTube	YouTube	YouTube	News	News	YouTube
Number of Data	25	50	50	203	1,970	5,100
Total Video Duration (Hours)	1.0	3.5	1.3	7.1	44.2	1229.9
Average Video Duration (mins)	2.4	3.9	1.6	2.1	1.4	14.5
Max Video Duration (mins)	5.4	10.8	3.5	6.9	4.8	115.4
Min Video Duration (mins)	0.5	1.4	0.8	0.3	0.4	1.0
Total Number of Text Tokens	–	–	–	0.2M	1.3M	11.2M
Avg. Keyframes per video	44	70	9.6	7.1	2.9	7.8
Avg. Text Summary Length	–	–	–	29.7	59.6	21.69
Number of Classes	25	10	7	–	–	170

personal style, clothes, sports, house, food, holiday, transportation, and hobbies. Each main category is further divided into 10 subcategories based on the popularity of Wikipedia, resulting in a total of 170 subcategories. To illustrate the subcategories associated with each main category, please refer to Figure 5.3. To ensure the dataset’s representativeness and practicality, we imposed certain criteria for video inclusion. Specifically, we only collected videos that were longer than 1 minute in duration while also ensuring that the maximum video duration did not exceed 120 minutes. Adhering to these guidelines allows a balance between capturing sufficient content in each video and preventing excessively lengthy videos from dominating the dataset. In total, our dataset comprises 170 subcategories and a grand total of 5,100 videos, all carefully selected to encompass a wide range of topics and characteristics.

5.3.2 Statistics of the Dataset

Figure 5.4 presents a comprehensive analysis of the MMSum dataset’s statistics. Figure 5.4(a) delves into the distribution of video durations, revealing the average duration spans approximately 15 minutes. In Figure 5.4(b), we show the distribution of the number of segments per video. The

graph in Figure 5.4(c) captures the distribution of segment durations, showcasing an intriguing resemblance to the Gaussian distribution with an approximate mean of 80 seconds. Figure 5.4(d) shows the distribution of the number of words per sentence.

5.3.3 Comparison with Existing Datasets

Table 5.2 presents a comparison between our MMSum dataset and existing video datasets. In contrast to standard video summarization datasets such as SumMe [143], TVSum [431], and OVP [20], our dataset, MMSum, stands out in several aspects. Firstly, the existing datasets lack textual data, whereas MMSum incorporates both video and textual information. Additionally, while the number of videos in SumMe, TVSum, and OVP is under 50, MMSum contains a substantial collection of 5,100 videos. Furthermore, the average duration of the videos in the aforementioned datasets is less than 4 minutes, whereas the videos in MMSum have an average duration of 14.5 minutes. Moreover, MMSum provides a significantly larger number of segments/keyframes per video compared to these standard datasets, making it more suitable for real-world applications. Comparing MMSum with other MSMO datasets like CNN and Daily Mail [115], we find that our dataset first surpasses them in terms of the number of videos. Furthermore, CNN and Daily Mail datasets were not curated based on specific classes; instead, the data was randomly downloaded, resulting in a lack of representativeness. In contrast, MMSum was carefully designed with 17 main categories and 170 subcategories, making it highly representative and practical. Although there are other MSMO datasets like VMSMO [295], we did not include them in the comparison table due to a large portion of the video links no longer be valid. Therefore, MMSum stands out as a comprehensive and reliable dataset for multimodal summarization tasks. The key distinguishing features of MMSum can be summarized as follows:

- MMSum offers an extensive and large-scale dataset, comprising an impressive collection of 5,100 human-annotated videos.
- The dataset showcases a remarkable range of untrimmed videos, varying in duration from concise 1-minute clips to extensive recordings spanning up to 115 minutes. This diversity allows for a comprehensive exploration of different video lengths and content complexities.
- MMSum’s strength lies in its meticulously crafted main category and subcategory groups, which exhibit an exceptional level of richness and granularity. With a keen focus on real-world applicability, these categories are thoughtfully designed to encapsulate the diverse facets and contexts of video data, ensuring relevance across a wide array of domains.
- To guarantee the highest quality and integrity of the dataset, MMSum undergoes rigorous manual verification. This meticulous process ensures that all modalities and information within the dataset are accurately annotated and readily accessible.

5.4 Angle II: Benchmark

5.4.1 Problem Formulation

The formulation of the MSMO task can be expressed as follows. A video and its corresponding transcripts are denoted as a pair (V, X) . The video input, represented by V , consists of a sequence

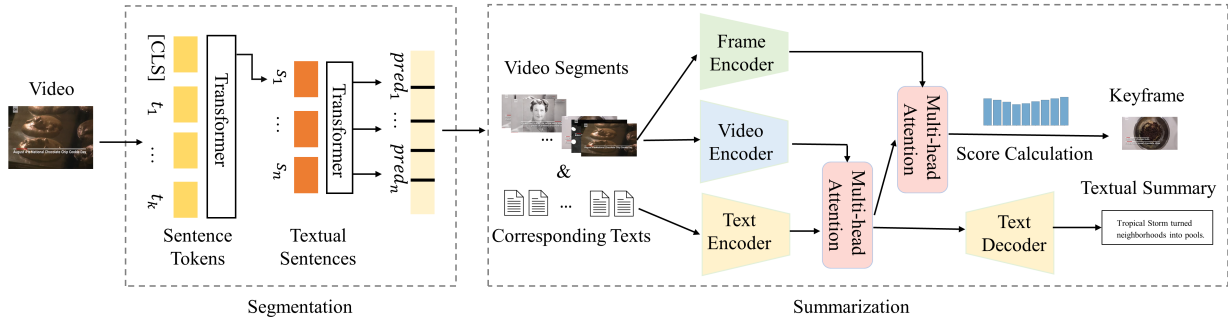


Figure 5.5: Our model comprises two modules: the segmentation module and the summarization module.

of frames: $V = (v_1, v_2, \dots, v_N)$. The corresponding transcripts, denoted as X , are a sequence of sentences: $X = (x_1, x_2, \dots, x_M)$. Note that M may not equal N due to one sentence per frame is not guaranteed in real-world videos. It is assumed that each video has a sequence of ground-truth textual summary, denoted as $Y = (y_1, y_2, \dots, y_L)$, and a sequence of ground-truth keyframe represented by $P = (p_1, p_2, \dots, p_L)$, where L is the number of segments. The objective of the MSMO task is to generate textual summaries \hat{Y} that capture the main points of the video and select keyframes \hat{P} from V to be the visual summaries.

5.4.2 Existing Methods

In order to conduct a thorough performance evaluation, we selected a set of established methods as our baselines. These baselines are chosen based on the public availability of official implementations, ensuring reliable and reproducible results. The selected baseline methods encompass:

- For Video Temporal Segmentation: Histogram Intersect [232], Moment Invariant [186], Twin Comparison [549], PySceneDetect [47], and LGSS [375].
- For Video Summarization: Uniform Sampling [190], K-means Clustering [151], VSUMM [80], and Keyframe Extraction [190].
- For Text Summarization: BERT2BERT [464], BART [235] (BART-large-CNN and BART-large-XSUM), Distilbart [420], T5 [369], Pegasus [550], and LED [33].

However, due to the absence of publicly available implementations for MSMO methods in the existing literature, there are no suitable methods that can be used as MSMO baselines.

5.4.3 Proposed Method

To solve the problem mentioned above and provide a MSMO baseline for the collected MM-Sum dataset, we propose a novel and practical approach to augment the MSMO baseline. Our method, which we have made accessible on our website, comprises two modules: segmentation and summarization. Our model is shown in Figure 5.5.

Segmentation Module

The primary objective of the segmentation module is to partition a given video into smaller segments based on the underlying content. This module operates by leveraging the entire transcript associated

with the video, employing a contextual understanding of the text. For the segmentation module, we adopted a hierarchical BERT architecture, which has demonstrated state-of-the-art performance [275]. It comprises two transformer encoders. The first encoder focuses on sentence-level encoding, while the second encoder handles paragraph-level encoding. The first encoder encodes each sentence independently using BERT_{LARGE} and then feeds the encoded embeddings into the second encoder. Notably, all sequences commence with a special token [CLS] to facilitate encoding at the sentence level. If a segmentation decision is made at the sentence level, the [CLS] token is utilized as input for the second encoder, which enables inter-sentence relationships to be captured through cross-attention mechanisms. This enables a cohesive representation of the entire transcript, taking into account the contextual dependencies between sentences.

Summarization Module

Upon segmenting the video, each video segment becomes the input to the summarization module. In line with the model architecture proposed in [223], we construct our summarization module. The summarization module incorporates three main encoders: a frame encoder, a video encoder, and a text encoder. These encoders are responsible for processing the video frames, video content, and corresponding text, respectively, to extract relevant feature representations. Once the features have been extracted, multi-head attention is employed to fuse the learned features from the different encoders, which allows for the integration of information across the modalities, enabling a holistic understanding of the video and its textual content. Following the fusion of features, a score calculation step is performed to select the keyframe, identifying the most salient frame within each video segment. Additionally, a text decoder is utilized to generate the textual summary, leveraging the extracted features and the fused representations.

Text Encoder The Transformer encoder [468] is employed to convert the text into a sequence of token embeddings. Inspired by [223, 537], we initialize the encoder’s weights using the pre-trained mT5 model [518]. To investigate the impact of task-specific pre-training, we fine-tune mT5 on the text-to-text summarization task, where $X_{enc} = \text{TextEncoder}(X)$.

Video Encoder To capture short-term temporal dependencies, we utilize 3D convolutional networks as in [223]. We partition the video into non-overlapping frame sequences and employ a 3D CNN network for feature extraction. Specifically, we utilize two different feature extractors. Firstly, we utilize the R(2 + 1)D model trained by [124] for video action recognition on weakly-supervised social-media videos. Secondly, we utilize the visual component of the S3D Text-Video model trained in a self-supervised manner by [288] on the HowTo100M dataset [290]. To incorporate long-term temporal dependencies, we process the sequence of video features using a Transformer encoder. This enables us to effectively capture and model the relationships between video frames over an extended duration, where $V_{enc} = 3D - CNN(V)$, $V_{enc} = \text{VideorEncoder}(V_{enc})$.

Frame Encoder To facilitate the selection of a specific frame as a cover picture, we require frame-level representations [223]. In our experimental setup, we sample one frame per second from the video. For feature extraction, we employ two models: EfficientNet [451] and Vision Transformer (ViT) [92]. Both models were pre-trained on the ImageNet dataset [395] for image classification tasks. To provide contextual information, we process the sequence of frame features

using a Transformer encoder, which captures the relationships and dependencies between the frame-level representations, enabling a more comprehensive understanding of the video content. Before applying the Transformer encoder, we ensure that both the video features and frame features have the same dimensions as the hidden states of the text encoder. In the case of a single model, the two sets of features are concatenated together before undergoing the projection step.

$$V_{\text{frame}} = \text{CNN}(\text{Sample}(V)), V_{\text{frame}} = \text{FrameEncoder}(V_{\text{frame}}) \quad (5.1)$$

Multi-head Attention In line with the study conducted by [223, 537], which explored various methods of integrating visual information into pre-trained generative language models, we adopt the approach of multi-head attention-based fusion. This technique allows us to obtain a vision-guided text representation by incorporating visual information into the model. The fusion process takes place after the last encoder layer, ensuring that both textual and visual inputs are combined effectively to enhance the overall representation.

$$\begin{aligned} Q &= X_{\text{enc}}W_q, Q \in \mathbb{R}^{M \times d}, K = V_{\text{enc}}W_k, K \in \mathbb{R}^{N' \times d} \\ V &= V_{\text{enc}}W_v, V \in \mathbb{R}^{N' \times d}, \tilde{X}_{\text{enc}} = \text{MHA}(Q, K, V), \tilde{X}_{\text{enc}} \in \mathbb{R}^{M \times d} \end{aligned} \quad (5.2)$$

As recommended by [223, 264], we incorporate the use of the forget gate mechanism (FG) in our model. This mechanism enables the model to filter out low-level cross-modal adaptation information. By utilizing the forget gate, our model can selectively retain and focus on the most relevant and informative features, disregarding less important or noisy information during the cross-modal fusion process. This helps improve the overall performance and robustness of the model in handling multimodal data.

$$\hat{X}_{\text{enc}} = \text{FG}(X_{\text{enc}}, \tilde{X}_{\text{enc}}), \hat{X}_{\text{enc}} \in \mathbb{R}^{M \times d} \quad (5.3)$$

To obtain the text+video guided frame representations, we employ the same multi-head attention mechanism. However, in this case, we substitute the input X_{enc} with V_{frame} and V_{enc} with \hat{X}_{enc} . By using the video frame features V_{frame} and the transformed text representations \hat{X}_{enc} , we generate the guided frame representations \hat{V}_{frame} through the multi-head attention process. This allows us to effectively incorporate both textual and visual information, guiding the frame-level representations based on the context provided by the text and video.

Text Decoder To generate the textual summary, we employ a standard Transformer decoder, initializing its weights with the mT5 checkpoint. The vision-guided text representation \hat{X}_{enc} serves as the input to the decoder. During training, we utilize the standard negative log-likelihood loss (NLLLoss) with respect to the target sequence Y . This loss function measures the dissimilarity between the predicted summary generated by the model and the ground truth summary, allowing the model to learn and improve its summary generation capabilities through backpropagation.

$$\hat{Y} = \text{TransformerDecoder}(\hat{X}_{\text{enc}}), \mathcal{L}_{\text{text}} = \text{NLLLoss}(\hat{Y}, Y) \quad (5.4)$$

To obtain the labels C for the cover picture (cover frame) selection, we calculate the cosine similarity between the CNN features of the reference cover picture and the candidate frames. In

most instances, the similarity values fall within the range of [0, 1], while the remaining negative values are mapped to 0. Previous studies such as [245] and [114] considered the frame with the maximum cosine similarity as the ground truth (denoted as C_{\max}), while considering the other frames as negative samples. However, upon analyzing the cosine similarity patterns, we observed that some videos exhibit multiple peaks or consecutive sequences of frames with very similar scores, capturing still scenes. We recognized that this could potentially harm the model’s performance, as very similar frames might be labeled as both positive and negative examples. To address this issue, in addition to the binary labels C_{\max} , we introduce smooth labels denoted as C_{smooth} . These smooth labels assign to each frame its cosine similarity score with the reference cover picture. By incorporating the smooth labels, we aim to provide a more nuanced and continuous representation of the frame similarities, allowing the model to learn from a broader range of similarity scores during the training process.

In our approach, we utilize a projection matrix to map the text+video guided frame representations $\widehat{V}_{\text{frame}}$ to a single dimension. This dimension reduction step allows us to obtain a compact representation of the frame features. Subsequently, we train the model using the binary cross-entropy (CE) loss, where the target labels C can either be C_{\max} or C_{smooth} . To train the entire model in an end-to-end fashion, we minimize the sum of losses \mathcal{L} , which includes the negative log-likelihood loss for textual summary generation and the binary cross-entropy loss for cover picture selection. By jointly optimizing these losses, the model learns to generate accurate summaries and make effective cover picture selections based on the input text and video. Please note that \mathcal{L} refers to the combined loss function that encompasses both the negative log-likelihood loss for summary generation and the binary cross-entropy loss for cover picture selection.

$$\widehat{C} = \widehat{V}_{\text{frame}} W_p, W_p \in \mathbb{R}^{d \times 1}, \mathcal{L}_{\text{image}} = \text{CE}(\widehat{C}, C), \mathcal{L} = \mathcal{L}_{\text{text}} + \mathcal{L}_{\text{image}} \quad (5.5)$$

5.5 Angle III: Tasks and Results

5.5.1 Types of tasks

Within our dataset, a wealth of information is available, enabling the exploration of various downstream tasks. These tasks encompass video summarization (VS), text summarization (TS), and multimodal video summarization with multimodal output (MSMO). For the train/val/test split, since our dataset is already randomly collected from YouTube, we designate the last 30% of videos within each subcategory (indexed 21-29) as the testing set. The remaining videos are then assigned to the training set (indexed 00-20) in each subcategory.

5.5.2 Evaluation of Traditional Tasks

Video Temporal Segmentation Evaluation For VTS, we followed [375] and adopted four common metrics: (1) Average Precision (AP); (2) F1 score; (3) M_{iou} : a weighted sum of the intersection of the union of a detected scene boundary with respect to its distance to the closest ground-truth scene boundary; and (4) Recall@ k s: recall at k seconds ($k = \{3, 5, 10\}$), the percentage of annotated scene boundaries which lies within k -second window of the predicted boundary.

Table 5.3: Comparison of video temporal segmentation results.

Model	Average Precision (AP) \uparrow	F1 \uparrow	M_{iou} \uparrow	Recall@3s \uparrow	Recall@5s \uparrow	Recall@10s \uparrow
Histogram Intersect	0.142	0.153	0.221	0.168	0.216	0.296
Moment Invariant	0.081	0.089	0.164	0.101	0.129	0.177
Twin Comparison	0.133	0.140	0.208	0.150	0.193	0.266
PySceneDetect	0.135	0.124	0.211	0.119	0.152	0.199
LGSS	0.243	0.352	0.216	0.163	0.216	0.272
Ours	0.503	0.423	0.223	0.325	0.341	0.366

Table 5.4: Comparison of video summarization results (whole-video setting and segment-level setting).

Setting	Model	RMSE \downarrow	PSNR \uparrow	SSIM \uparrow	SRE \downarrow	Precision \uparrow	Recall \uparrow	F1 Score \uparrow
Whole-video	Uniform [190]	0.479	4.044	0.076	49.808	0.077	0.100	0.049
	K-means [151]	0.348	8.234	0.055	46.438	0.072	0.182	0.103
	VSUMM [80]	0.279	9.226	0.053	44.862	0.054	0.259	0.088
	Ours	0.112	25.280	0.697	23.550	0.320	0.290	0.321
Segment-level	Uniform [190]	0.237	6.307	0.085	42.495	0.186	0.179	0.105
	K-means [151]	0.167	10.123	0.144	46.533	0.123	0.172	0.143
	VSUMM [80]	0.122	18.818	0.258	41.601	0.160	0.207	0.171
	Ours	0.091	36.370	0.698	23.430	0.333	0.275	0.255

Video Summarization Evaluation The quality of the chosen keyframe is evaluated by Root Mean Squared Error (RMSE), Structural Similarity Index (SSIM), Signal reconstruction error ratio (SRE), and Spectral angle mapper (SAM), between image references and the extracted video frames [301]. In addition, we also adopted precision, recall, and F1 score based on SSIM for evaluation.

Text Summarization Evaluation The quality of generated textual summary is evaluated by standard evaluation metrics, including BLEU [329], METEOR [84], ROUGE-L [257], CIDEr [469], and BertScore [554], following previous works [60, 295, 411]. ROUGE-1, ROUGE-2, and ROUGE-L refer to the overlap of unigram, bigrams, and the longest common subsequence between the decoded summary and the reference, respectively [257].

5.5.3 Results and Discussion

Supervised training leads to more accurate video temporal segmentation results The performance of video temporal segmentation has a great impact on the final performance, so in this section, we compare the performance of VTS with several baselines: Histogram Intersect [232], Moment Invariant [186], Twin Comparison [549], PySceneDetect [47], and LGSS [375]. The results, displayed in Table ??, indicate that LGSS outperforms the other baselines but falls short when compared to our model. Both our method and LGSS are trained using a supervised approach, which leads to improved performance compared to unsupervised baselines. Moreover, our approach incorporates attention mechanisms, potentially contributing to better results.

Supervised methods outperform unsupervised methods on video summarization In our video summarization study, we have chosen the following methods as our baseline comparisons: Uniform Sampling [190], K-means Clustering [151], and VSUMM [80]. The results, presented in Table 5.4, are under various evaluation metrics. For RMSE and SRE, lower values indicate better performance,

Table 5.5: Comparison of textual summarization results (whole-video setting and segment-level setting).

Setting	Model	BLEU-1 \uparrow	ROUGE-1 \uparrow	ROUGE-2 \uparrow	ROUGE-L \uparrow	METEOR \uparrow	CIDEr \uparrow	SPICE \uparrow	BertScore \uparrow
Whole-video	BERT2BERT [464]	22.59	3.75	0.45	3.41	5.65	1.76	2.91	71.12
	BART-large-CNN [235]	29.17	3.19	0.51	3.04	2.99	2.28	11.27	68.84
	BART-large-XSUM [235]	30.91	3.83	0.57	3.59	3.99	2.56	3.71	69.56
	Distilbart [420]	26.46	3.87	3.87	0.47	3.59	2.25	4.16	69.37
	T5 [369]	25.39	3.51	0.43	3.21	4.51	1.97	5.66	70.38
	Pegasus [550]	26.73	3.75	0.52	3.40	4.52	2.38	7.82	68.92
	LED [33]	26.47	3.81	0.25	3.51	3.45	1.78	6.72	68.45
	Ours	32.61	9.41	2.86	9.15	4.01	4.01	10.11	74.46
Segment-level	BERT2BERT [464]	13.58	4.70	1.95	4.53	28.59	11.73	10.13	71.76
	BART-large-CNN [235]	22.79	6.45	2.46	6.32	26.21	20.64	10.13	71.44
	BART-large-XSUM [235]	20.89	7.31	2.77	7.13	29.36	20.90	10.20	71.42
	Distilbart [420]	14.77	1.95	0.15	1.87	23.52	11.83	10.53	66.46
	T5 [369]	16.48	6.17	3.03	5.99	28.22	20.96	10.35	71.95
	Pegasus [550]	16.17	3.41	0.96	3.29	29.82	17.26	10.39	67.81
	LED [33]	16.03	3.80	0.60	3.64	29.81	15.85	10.99	68.46
	Ours	23.36	13.61	4.58	13.24	30.01	21.06	10.28	85.19

Table 5.6: Comparison of MSMO results.

Methods	Text					Video				
	BLEU \uparrow	METEOR \uparrow	CIDEr \uparrow	SPICE \uparrow	BertScore \uparrow	PSNR \uparrow	SSIM \uparrow	Precision \uparrow	Recall \uparrow	F1 Score \uparrow
LGSS + VSUMM + T5	27.35	24.32	3.94	5.57	62.77	16.234	0.198	0.143	0.152	0.147
LGSS + VSUMM + BART-large-XSUM	24.83	24.12	3.97	8.86	39.20	16.234	0.198	0.143	0.152	0.147
LGSS + VSUMM + BERT2BERT	13.26	24.83	3.68	9.23	64.34	16.234	0.198	0.143	0.152	0.147
LGSS + VSUMM + BART-large-CNN	24.93	28.61	3.78	9.84	64.44	16.234	0.198	0.143	0.152	0.147
Ours	33.36	30.31	4.06	10.28	85.19	36.370	0.298	0.233	0.275	0.155

whereas, for the remaining metrics, higher values are desirable. From Table 5.4, we can observe that VSUMM showcases the strongest performance among the baseline methods, yet it still falls short compared to our proposed method. But we can conclude that supervised methods outperform unsupervised methods.

Pretrained large language models can still do well in text summarization In the context of textual summarization, we have considered a set of representative models as our baseline comparisons: BERT2BERT [464], BART [235] (including BART-large-CNN and BART-large-XSUM), Distilbart [420], T5 [369], Pegasus [550], and Longformer Encoder-Decoder (LED) [33]. The performance of these models is summarized in Table 5.5. Among the baselines, T5, BART-large-XSUM, BART-large-CNN, and BERT2BERT exhibit superior performance, with T5 demonstrating relatively better results across various text evaluation metrics. In addition, the ROUGE score may not effectively capture performance differences compared to other evaluation metrics, because ROUGE does not take into account the semantic meaning and the factual accuracy of the summaries.

MSMO results may depend on segmentation results and summarization methods In the field of MSMO, we encountered limitations in accessing the codebases of existing works such as [52, 114, 115, 172, 542, 576]. Therefore, we independently implemented several baselines to evaluate their performance on the MMSum dataset. For this purpose, we utilized LGSS as the segmentation backbone, VSUMM as the video summarizer, and selected text summarizers that exhibited the best performance in text summarization. The results are presented in Table 5.6. Based on the findings, it is evident that the aforementioned combination approaches still fall short in comparison to our proposed method. This also indicates that the accuracy of temporal segmentation

is crucial prior to generating summaries, highlighting it as a critical step and task preceding MSMO.

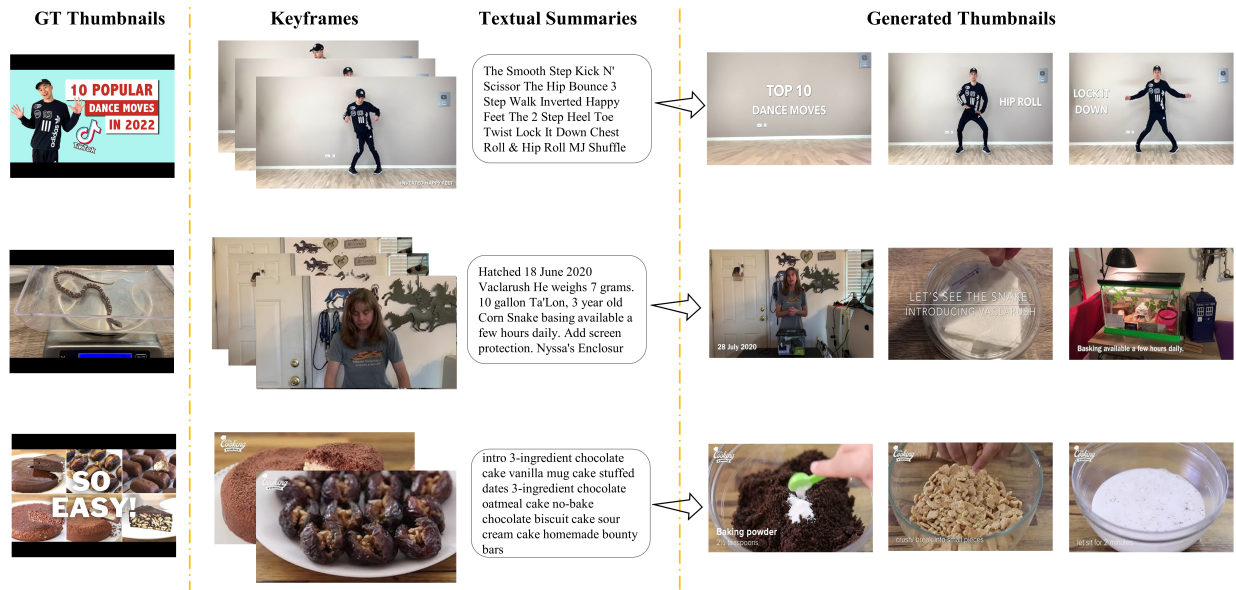


Figure 5.6: Comparison of GT thumbnails and our generated ones.

5.5.4 Thumbnail Generation

One direct and practical application of the MSMO task is to automatically generate thumbnails for a given video, which has become increasingly valuable in various real-world applications. With the exponential growth of online videos, effective and efficient methods are required to extract visually appealing and informative thumbnail representations. In addition, many author-generated thumbnails involve words or titles that describe the whole video to attract more users. In the context of online platforms, such as video-sharing websites or social media platforms, compelling thumbnails can significantly impact user engagement, content discoverability, and overall user experience. The benefits of automated thumbnail generation extend beyond user engagement and content discoverability. In e-commerce, for instance, thumbnails can play a vital role in attracting potential buyers by effectively showcasing products or services. Similarly, in video editing workflows, quick and accurate thumbnail generation can aid content creators in managing and organizing large video libraries efficiently.

In our setting, we take advantage of the results by MSMO, which contain both visual summaries and text summaries and combine them to generate thumbnails for a given video. In summary, the selected keyframes and generated textual summaries from the MSMO task are subsequently utilized to create the thumbnail. To ensure an aesthetically pleasing appearance, we randomly sample from a corpus of fonts from Google Fonts and font sizes to utilize in the generated thumbnails. Moreover, a random set of coordinates on the selected keyframe is sampled for the placement of the text. Finally, the text is pasted onto the keyframe from the outputted set of coordinates to complete thumbnail generation.

More specifically, the font is randomly selected from 100 fonts, and the size of the font varies by 175 font sizes. Here we list 20 examples of fonts we used in our experiments: [Roboto, Open

Sans, Lato, Montserrat, Raleway, Oswald, Source Sans Pro, Poppins, Noto Sans, Roboto Slab, Merriweather, Ubuntu, PT Sans, Playfair Display, Fira Sans, Nunito, Roboto Condensed, Zilla Slab, Arvo, Muli]. We randomly select one font and a random font size. Given the image size of the selected keyframes, we also randomly select coordinates for where the text should be pasted onto the selected keyframes. We then paste the generated textual summary, which is modified by the randomly selected font and font size, onto the selected keyframes. Some examples are shown in Figure 5.6.

Limitations and Future Work Directions The lack of publicly available MSMO baselines in existing literature underscores a significant gap, emphasizing the need for future efforts in this area. Advancing the field requires tackling the complex task of creating a diverse and extensive collection of baselines.

Despite the progress made in automated thumbnail generation, challenges remain. These include enhancing the accuracy of thumbnail selection, accommodating various video genres and content types, and taking into account user preferences and context-specific requirements.

Moreover, addressing ethical concerns related to potential biases, representation, and content moderation is crucial to ensuring fair and inclusive thumbnail generation. Exploring new quantitative evaluation metrics for the thumbnail generation task could also pave the way for valuable advancements in this domain.

5.6 Conclusion

In this chapter, our main goal is to overcome the limitations of existing MSMO datasets by creating a comprehensive dataset called MMSum. Our contributions are summarized as follows.

- MMSum was meticulously curated to ensure top-notch quality of MSMO data, making it a valuable resource for tasks like video temporal segmentation, video summarization, text summarization, and multimodal summarization.
- Additionally, we introduced a novel benchmark based on the MMSum dataset. This benchmark enables researchers and practitioners to assess their algorithms and models across a range of tasks.
- Moreover, leveraging the results from MSMO, we introduced a new task: automatically generating thumbnails for videos. This innovation has the potential to significantly enhance user engagement, content discoverability, and overall user experience.
- Our MMSum dataset can be found at <https://mmsum-dataset.github.io/>.
- [357] has 24 stars, 16 clones, and 424 viewers (as of March 24, 2024).

Part II

Robustness of Multimodal Models under Perturbations

Chapter 6

Robustness of Multimodal Models under Distribution Shift

In the previous chapters, we show that multimodal models have shown remarkable performance. However, evaluating robustness against distribution shifts is crucial before adopting them in real-world applications. In this chapter, we investigate the robustness of 9 popular open-sourced image-text models under common perturbations on five tasks (image-text retrieval, visual reasoning, visual entailment, image captioning, and text-to-image generation). In particular, we propose several new multimodal robustness benchmarks by applying 17 image perturbation and 16 text perturbation techniques on top of existing datasets. We observe that multimodal models are not robust to image and text perturbations, especially to image perturbations. Among the tested perturbation methods, character-level perturbations constitute the most severe distribution shift for text, and zoom blur is the most severe shift for image data. We also introduce two new robustness metrics (MMI and MOR) for proper evaluations of multimodal models. We hope our extensive study sheds light on new directions for the development of robust multimodal models. The MMRobustness evaluation benchmark and codebase can be found at: <https://MMRobustness.github.io>.

6.1 Introduction

Many multimodal learning datasets and models have been collected and proposed to accelerate research in this field [8, 62, 93, 120, 212, 241, 242, 248, 251, 367, 367, 373, 480, 523, 536, 553]. Despite the extraordinary performance and exciting potential, we find that multimodal models are often vulnerable under distribution shifts. In Figure 6.1, we show interesting examples of image captioning under image perturbations using BLIP [241], and text-to-image generation under text perturbations using Stable Diffusion [389]. For image captioning, we observe that by simply adding noise, blur, or pixelation to the original image, the generated captions become incorrect. For text-to-image generation, applying keyboard typos, OCR errors, or synonym replacements to the original sentence, can lead to generated images containing incomplete visual information.

There is a sizable literature on robustness evaluation of unimodal vision models [10, 37, 88, 94, 132, 280, 284, 335, 497, 533, 560, 563] or unimodal language models [48, 91, 127, 137, 282, 396, 425, 475, 482, 486]. Several recent work [78, 108, 119, 128, 319] have unsystematically tested or probed a few pre-trained multimodal models, including CLIP [367] and DALL-E 2 [373]. However, the robustness evaluation of multimodal image-text models under distribution shift has rarely been

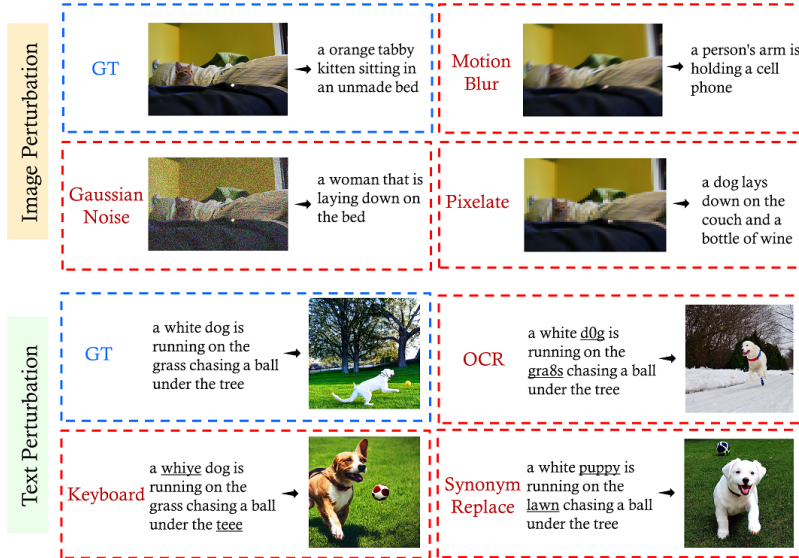


Figure 6.1: Multimodal models are sensitive to image/text perturbations (original image-text pairs are shown in blue boxes, perturbed ones are in red). Image captioning (Top): Adding image perturbations can result in incorrect captions, e.g., the tabby kitten is mistakenly described as a woman/dog. Text-to-image generation (bottom): Applying text perturbations can result in the generated images containing incomplete visual information, e.g., the tree is missing in the two examples above.

studied. To our best knowledge, there is currently no benchmark dataset nor a comprehensive study of how the perturbed data can affect their performance. Hence in this chapter:

- We build multimodal robustness evaluation benchmarks by leveraging existing datasets and tasks, e.g., image-text retrieval (Flicker30K, COCO), visual reasoning (NLVR2), visual entailment (SNLI-VE), image captioning (COCO), and text-to-image generation (COCO). We analyze the robustness of 9 multimodal models under distribution shifts, which include 17 image perturbation and 16 text perturbation methods.
- We introduce two new robustness metrics, one termed MMI (MultiModal Impact score), to account for the relative performance drop under distribution shift in 5 downstream applications. The other one is named MOR (Missing Object Rate), which is based on open-set language-guided object detection and the first object-centric metric proposed for text-to-image generation evaluation.
- We find that multimodal image-text models are more sensitive to image perturbations than text perturbations. In addition, *zoom blur* is the most effective attack for image perturbations, while character-level perturbations show a higher impact than word-level and sentence-level perturbations for text. In addition, we provided interpretations of performance drop by different perturbation methods using Optimal Transport alignment and attention.

6.2 Related Work

Robustness of unimodal vision models is a longstanding and challenging goal of computer vision [533]. Stable training, adversarial robustness, out-of-distribution, transfer learning, and many

other aspects have been studied by previous works in deep learning era [88, 94, 132, 560]. Recently, several studies have shown that Vision Transformer (ViT) [92] tend to be more robust than previous models, e.g., work that studied the robustness against common corruptions and perturbations [37], robustness for distribution shifts and natural adversarial examples [335], robustness against different Lp-based adversarial attacks [280], adversarial examples [284], and adaptive attacks [10]. Several robustness benchmarks have been proposed, e.g., ImageNet-C and ImageNet-P [158], Stylized-ImageNet [123], ImageNet-A and ImageNet-O [161], ImageNet-V2 [378]. Recently, [497] conducted a large-scale robustness study based on natural distribution shifts. [140] built the GRIT benchmark to evaluate the performance, robustness, and calibration of a vision system across different image tasks.

Robustness of unimodal language models under distribution shift or adversarial attack has been explored by many works, i.e., [48, 486] provided reviews of how to define, measure and improve robustness of NLP systems, [482] proposed controlled adversarial text generation to improve robustness, [127] unified four standard evaluation paradigms, [425] proposed a search and semantically replace strategy, [91] studied robustness against word substitutions, [282] formalised the concept of semantic robustness, etc. For benchmarks, [159] systematically examined and measured the Out-of-Distribution (OOD) generalization for seven NLP datasets. [73] built a large benchmark and analyzed the impact of robustness under distribution shifts, calibration, OOD detection, fairness, privacy leakage, smoothness, and transferability. Recently, [299] presented empirical results achieved with a comprehensive set of non-adversarial perturbation methods for testing the robustness of NLP systems on non-synthetic text. [137] proposed a multilingual evaluation platform to provide comprehensive robustness analysis. [475] proposed a benchmark to evaluate the vulnerabilities of modern large-scale language models under adversarial attacks.

Robustness of Multimodal Models There is a sizable literature on robustness evaluation of unimodal vision models [10, 37, 88, 94, 132, 280, 284, 335, 533, 560, 563] or unimodal language models [48, 91, 127, 137, 282, 396, 425, 475, 482, 486]. However, robustness evaluation of multimodal image-text models under distribution shift has rarely been studied [78, 128]. Previous works [108, 119, 128, 319] have unsystematically tested some pre-trained models, i.e., CLIP [367], by attacking with text patches and adversarial pixel perturbations. [78] found that DALLE-2 [373] has a hidden vocabulary that can be used to generate images with absurd prompts. [102] found that diverse training distribution is the main cause for robustness gains. [64] studied the text-to-image generative models about visual reasoning skills and social bias. For benchmarks, [243] collected an Adversarial VQA dataset to evaluate the robustness of VQA models. [404] studied the robustness of video-text models under perturbations, but they only focused on one video-text retrieval task. In this chapter, we conduct a systematic robustness evaluation of recent multimodal image-text models on 5 different downstream tasks based on new datasets and metrics.

6.3 Multimodal Robustness Benchmark

Distribution shift is one of the significant problems of applying models in real-world scenarios [270, 455]. Distribution shift happens when the training data distribution $p_{tr}(\mathbf{x} | \mathbf{y})$ is different from the data distribution to which the model has applied at test time $p_{te}(\mathbf{x} | \mathbf{y})$. A model is said to

Table 6.1: Image perturbations.

Category	Perturbation	Description	Severities
Noise	Gaussian Noise	Gaussian noise can appear in low-lighting conditions.	5
	Shot Noise	Shot noise, also called Poisson noise, is electronic noise caused by the discrete nature of light itself.	5
	Impulse Noise	Impulse noise is a color analog of salt-and-pepper noise and can be caused by bit errors.	5
	Speckle Noise	Speckle noise is the noise added to a pixel that tends to be larger if the original pixel intensity is larger.	5
Blur	Defocus Blur	Defocus blur occurs when an image is out of focus.	5
	Glass Blur	Frosted Glass Blur appears with “frosted glass” windows or panels.	5
	Motion Blur	Motion blur appears when a camera is moving quickly.	5
	Zoom Blur	Zoom Blur occurs when a camera moves toward an object rapidly.	5
Weather	Snow	Snow is a visually obstructive form of precipitation.	5
	Frost	Frost forms when lenses or windows are coated with ice crystals.	5
	Fog	Fog shrouds objects and is rendered with the diamond-square algorithm.	5
	Brightness	Brightness varies with daylight intensity.	5
Digital	Contrast	Contrast can be high or low depending on lighting conditions and the photographed object’s color.	5
	Elastic	Elastic transformations stretch or contract small image regions.	5
	Pixelate	Pixelation occurs when upsampling a low-resolution image.	5
	JPEG Compression	JPEG is a lossy image compression format that introduces compression artifacts.	5
Stylize	Stylize	Stylized data is generated by transferring the style information to the content images by AdaIN style transfer [184].	5
Sum	17	—	85

be robust on the OOD data, if it still produces accurate predictions on the test data. To evaluate the robustness of large pretrained multimodal models under distribution shift, we start by building several evaluation benchmark datasets via perturbing the original image-text pairs on either the image side or text side. We use these perturbations to simulate distribution shifts of various intensities and use them to stress-test the robustness of the given models.

6.3.1 Image Perturbation

To simulate distribution shifts for the image data, we adopt the perturbation strategies from ImageNet-C [158] and Stylize-ImageNet [123, 287]. We include Stylize-ImageNet for its effectiveness in perturbing the original image by breaking its shape and texture [123]. The perturbations are grouped into five categories: **noise**, **blur**, **weather**, **digital**, and **stylize**. As shown in Table 6.1, we use 17 image perturbation techniques, (1) Noise: *Gaussian noise*, *shot noise*, *impulse noise*, *speckle noise*; (2) Blur: *defocus blur*, *glass blur*, *motion blur*, *zoom blur*; (3) Weather: *snow*, *frost*, *fog*, *brightness*; (4) Digital: *contrast*, *elastic*, *pixelate*, *JPEG compression*; and (5) *stylize*. Note that real-world corruptions can manifest themselves at varying intensities, we thus introduce variation for each corruption following [123, 158, 287]. In our benchmark, each category has five levels of severity, resulting in 85 perturbation methods in total. Note that these strategies are commonly considered

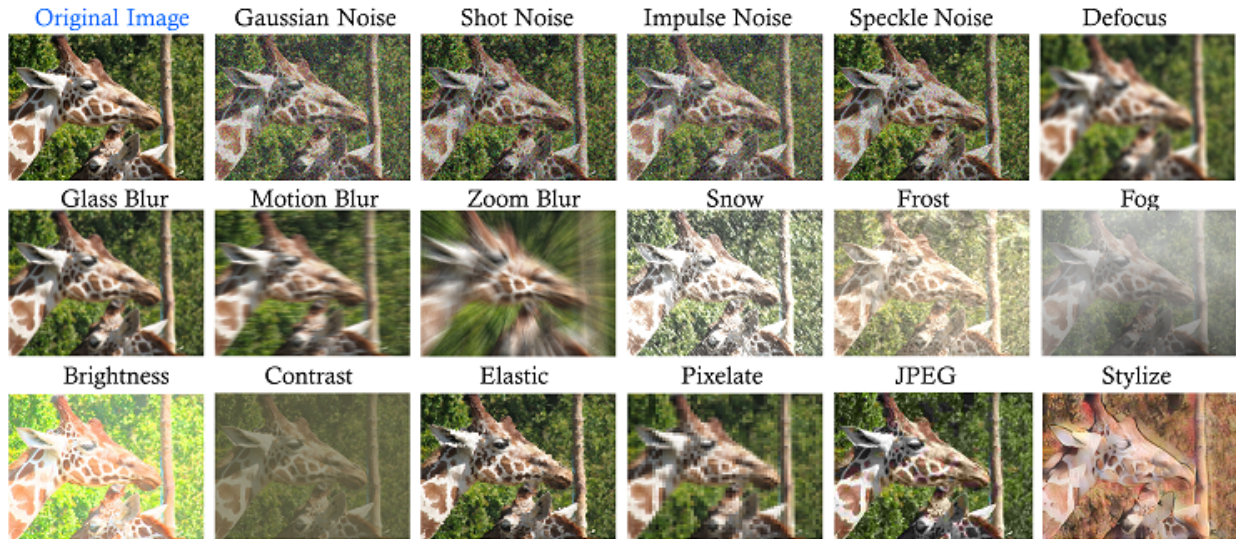


Figure 6.2: Examples of our 17 image perturbations. The original image is taken from the COCO dataset and shown on the top left.

synthetic distribution shifts and can serve as a good starting point since they are precisely defined and easy to apply. Examples of the perturbed images can be seen in Figure 6.2.

6.3.2 Text Perturbation

To simulate the distribution shifts in language, we design 16 text perturbation techniques grouped into three categories: **character-level**, **word-level**, and **sentence-level**. In detail, as in Table 6.2, for character-level perturbation, we adopt 6 strategies from [277], including *keyboard*, *OCR*, *character insert (CI)*, *character replace (CR)*, *character swap (CS)*, *character delete (CD)*. These perturbations can be considered as simulating real-world typos or mistakes during typing. For word-level perturbation, we adopt 5 strategies from EDA and AEDA [205, 495], including *synonym replacement (SR)*, *word insertion (WR)*, *word swap (WS)*, *word deletion (WD)*, and *insert punctuation (IP)*. These perturbations aim to simulate different writing habits that people may replace, delete, or add words to express the same meaning. For sentence-level perturbation, (1) we first adopt the style transformation strategies from [100, 238, 404, 405], i.e., transferring the style of text into *formal*, *casual*, *passive*, and *active*; (2) we also adopt the *back translation* method from [277]. These perturbations will focus more on language semantics due to the differences in speaking or writing styles or translation errors. Similar to image perturbations, we introduce severity levels to each strategy. For strategies within the character-level and word-level perturbations, we apply 5 severity levels similar to image perturbations, while for strategies within the sentence-level perturbations, there is only one severity level. This leads to a total of 60 text perturbation methods. We emphasize that these perturbation techniques cover some of the actual text distribution shifts we encounter in real-world applications (e.g., typos, word swaps, style changes, etc.). Models for text data that are deployed in real-world settings need to be robust with respect to these perturbations. Examples of the text perturbations are shown in Table 6.3.

Table 6.2: Text perturbations.

Category	Perturbation	Description	Severities
Character-level	Keyboard	Substitute character by keyboard distance with probability p .	5
	OCR	Substitute character by pre-defined OCR error with probability p .	5
	Character Insert (CI)	Insert character randomly with probability p .	5
	Character Replace (CR)	Substitute character randomly with probability p .	5
	Character Swap (CS)	Swap character randomly with probability p .	5
	Character Delete (CD)	Delete character randomly with probability p .	5
Word-level	Synonym Replacement (SR)	Randomly choose n words from the sentence that are not stop words. Replace each of these words with one of its synonyms chosen at random.	5
	Word Insertion (WI)	Find a random synonym of a random word in the sentence that is not a stop word. Insert that synonym into a random position in the sentence. Do this n times.	5
	Word Swap (WS)	Randomly choose two words in the sentence and swap their positions. Do this n times.	5
	Word Deletion (WD)	Each word in the sentence can be randomly removed with probability p .	5
	Insert Punctuation (IP)	Random insert punctuation in the sentence with probability p .	5
Sentence-level	Formal	Transfer the text style to Formal.	1
	Casual	Transfer the text style to Casual.	1
	Passive	Transfer the text style to Passive.	1
	Active	Transfer the text style to Active.	1
	Back Translation	Translate source to German and translate it back to English via [315].	1
Sum	16	—	60

Fidelity To build a convincing benchmark, we need to ensure that the perturbed text has the same semantics as the original one. Otherwise, for image-text pairs in multimodal learning, the perturbed text will not match the original image and, hence, would no longer represent a meaningful image-text pair. In this chapter, we use paraphrases from pretrained sentence-transformers [382] to evaluate the semantic similarity between the original and the perturbed sentences. Specifically, “paraphrase-mpnet-base-v2” [382] is used to extract the original and perturbed sentence embeddings for computing similarity score α_s . Given a predefined tolerance threshold α_0 , a higher score $\alpha_s > \alpha_0$ means the perturbed text still has similar semantics with the original text. However, if $\alpha_s < \alpha_0$ indicating their semantics are different, we will perturb the sentence again until the semantic similarity score meets the requirement, in a reasonable looping time $N_{max} = 100$. Beyond N_{max} , we will remove this text sample from our robustness benchmark. This procedure guarantees semantic closeness and ensures our perturbed data could serve as a valid evaluation benchmark for multimodal image-text models.

6.4 Experiments

Using our multimodal robustness benchmark, we are able to answer the following questions: **(1)** How robust are multimodal pretrained image-text models under distribution shift? **(2)** What is the sensitivity of each model under different perturbation methods? **(3)** Which model architecture or loss objectives might be more robust under image or text perturbations? **(4)** Are there any particular

Table 6.3: Example of our 16 text perturbations. The original text is taken from the COCO dataset and denoted as clean in the first row.

Category	Perturbation	Example
Original	Clean	An orange metal bowl strainer filled with apples.
Character	Keyboard	An orange metal bowl strainer filled with apples.
	OCR	An orange metal bowl strainer filled with apples.
	CI	An orange metal bowl strainer filled with apples.
	CR	An orange metal bowl strainer filled with apples.
	CS	An orange metal bowl strainer filled with apples.
	CD	An orange metal bowl strainer filled with apples.
	Word	SR
WI		An old orange metal bowl strainer filled with apples.
WS		An orange metal strainer bowl filled with apples.
WD		An orange metal bowl strainer [X] with apples.
IP		An orange metal bowl ? strainer filled with apples.
Sentence	Formal	An orange metal bowl strainer contains apples.
	Casual	An orange metal bowl is filled with apples.
	Passive	Some apples are in an orange metal bowl strainer.
	Active	There are apples in an orange metal bowl strainer.
	Back trans	Apples are placed in an orange metal bowl strainer.

image/text perturbation methods that can consistently show significant influence?

6.4.1 Evaluation Tasks, Datasets and Models

As shown in Table 6.4, we select five widely adopted downstream tasks for a comprehensive robustness evaluation under distribution shift, including image-text retrieval, visual reasoning (VR), visual entailment (VE), image captioning, and text-to-image generation. For each task, we perturb the corresponding datasets, i.e., Flickr30K [535], COCO [258], NLVR2 [441], and SNLI-VE [509, 510], using the image perturbation (IP) and text perturbation (TP) methods introduced in Sec. 6.3. This leads to our 8 benchmark datasets: (1) Flickr30K-IP, Flickr30K-TP, COCO-IP, and COCO-TP for image-text retrieval evaluation; (2) NLVR2-IP and NLVR2-TP for visual reasoning evaluation; (3) SNLI-VE-IP and SNLI-VE-TP for visual entailment evaluation; (4) COCO-IP for image captioning evaluation; and (5) COCO-TP for text-to-image generation evaluation. We select 9 representative large multimodal models, which have publicly released their code and pretrained weights: CLIP [367], ViLT [212], ALBEF [242], BLIP [241], TCL [523], METER [93], GRIT [316], GLIDE [317] and Stable Diffusion [389]. We appreciate the authors for making their models publicly available.

6.4.2 Evaluation Metrics

We adopt standard evaluation metrics for each task. To be specific, for image-text retrieval, we use recall and RSUM (i.e., the sum of recall R@K metric [503]). As for visual reasoning and visual entailment tasks, we use prediction accuracy. For image captioning, we use standard text evaluation metrics, i.e., BLEU [329], METEOR [84], ROUGE-L [257], and CIDEr [469]. For text-to-image

Table 6.4: Evaluation tasks, datasets, models and metrics used in our study.

Task	Datasets	Models	Evaluation metrics
Image-text Retrieval	Flicker30K, COCO	CLIP, ViLT, TCL, ALBEF, BLIP	Recall K (R@K), K = {1, 5, 10}, and RSUM
Visual Reasoning	NLVR2	ALBEF, ViLT, BLIP, TCL, METER	Prediction accuracy
Visual Entailment	SNLI-VE	ALBEF, TCL, METER	Prediction accuracy
Image Captioning	COCO	BLIP, GRIT	BLEU, METEOR, ROUGE-L, CIDEr
Text-to-image Generation	COCO	Stable Diffusion, GLIDE	FID, CLIP-FID, MOR (ours)

generation, we use FID [163] and CLIP-FID [226, 333] scores, and our proposed MOR (details will be introduced later) to evaluate the quality of the generated images.

MultiModal Impact score (MMI) To evaluate the robustness of a model, it is crucial to measure the relative performance drop between the In-Distribution (ID) and OOD performance. Recall the example given by [455], let d_1 be the ID dataset (where the model is trained), and d_2 be an OOD dataset, then a model m_1 should be considered more robust than model m_2 if m_1 ’s performance drop is less significant than m_2 when evaluated from d_1 to d_2 , even though m_2 ’s absolute accuracy/recall on d_2 may still be higher than m_1 ’s. To quantitatively measure the robustness of multimodal image-text models, we introduce a new robustness evaluation metric, termed **MultiModal Impact score (MMI)**. We compute MMI as the averaged performance drop compared with the non-perturbed performance (“clean”), i.e., $MMI = (s_c - s_p)/s_c$ where s_p is the perturbed score and s_c is the clean score. Here, the score can be any standard metric mentioned above, e.g., recall, RSUM, accuracy, FID, and CLIP-FID. In the following experiments, we report both the standard evaluation metrics on the perturbed (OOD) datasets as well as their corresponding MMI variants.

6.4.3 Robustness Evaluation under Distribution Shift

Image-text retrieval We present the evaluation results under image perturbations in Table 6.5 [Top] and results under text perturbations in Table 6.5 [Bottom]. For simplicity, we only report the RSUM scores here.

Inspecting Table 6.5 [Top], we observe that the performance of all models drops under image perturbation. Although different perturbation methods have various impacts on different models, we observe the following general trends. We find that most multimodal models are most sensitive to *zoom blur*. Additionally, we find that *glass blur* and *brightness* are the two “softest” perturbation methods, where the performance of all evaluated models deteriorates the least. Comparing the MMI score for both Flickr30K and COCO datasets, CLIP zero-shot (ZS) is more robust than other models, possibly due to it being trained on the large WIT400M dataset [367]. As indicated in [455], training models on large and diverse datasets often leads to increased robustness. For text perturbations in Table 6.5 [Bottom], we also find the performance of all models drop. In addition, we observe the following general trends. Character-level perturbations show more influence than word-level and sentence-level perturbations. In particular, *keyboard* and *character replace (CR)* consistently show a high impact on models’ robustness, while *insert punctuation (IP)*, *formal*, and *active* are the least effective text perturbations.

For both image and text perturbations, we see that BLIP shows the best robustness performance on two datasets, i.e., the lowest MMI score. We hypothesize that using an encoder-decoder architecture and generative language modeling objective in BLIP is helpful for image-text retrieval.

Table 6.5: **Image-text retrieval. [Top]** Robustness evaluations on Flickr30k-IP and COCO-IP. **[Bottom]** Robustness evaluations on Flickr30k-TP and COCO-TP datasets. We report averaged RSUM where the most effective perturbation results are marked in bold, and the least effective perturbation results are underlined. The MMI impact score is marked in blue; the lower, the better.

Dataset	Method	Noise					Blur				Weather				Digital			Stylize		ave	MMI
		Clean	Gauss.	Shot	Impulse	Speckle	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contrast	Elastic	Pixel	JPEG	Stylize		
Flickr30K	ViLT FT	522.0	413.0	419.6	396.9	387.1	417.6	489.0	388.4	236.3	332.7	453.1	455.8	<u>496.9</u>	372.2	461.7	277.4	487.6	387.1	408.7	↓ 21.7%
	CLIP ZS	533.7	501.7	504.2	481.2	515.5	502.1	<u>530.1</u>	509.7	457.8	470.7	495.6	519.7	<u>530.1</u>	515.4	510.4	469.5	524.6	447.6	499.2	↓ 6.5%
	CLIP FT	544.3	500.1	503.8	479.1	522.1	493.3	<u>536.9</u>	513.3	444.4	464.4	503.2	529.7	<u>543.5</u>	521.5	513.9	453.9	528.6	436.9	499.3	↓ 8.3%
	TCL ZS	563.8	464.9	467.0	458.4	498.0	429.8	506.6	388.5	251.3	407.3	449.5	434.2	<u>509.1</u>	473.2	434.4	247.2	502.2	343.4	427.4	↓ 24.2%
	TCL FT	573.4	529.9	532.6	527.7	551.6	504.5	<u>566.0</u>	513.9	397.3	521.7	551.0	554.1	568.0	557.1	421.0	372.0	555.4	448.7	516.2	↓ 10.0%
	ALBEF FT	577.7	533.8	538.3	532.0	557.8	528.8	569.2	516.0	416.1	532.0	558.1	560.4	<u>572.0</u>	550.6	538.7	435.9	559.8	464.1	527.3	↓ 8.7%
	BLIP FT	580.9	536.2	538.9	528.6	560.8	529.4	571.6	525.7	412.1	456.6	513.4	568.5	<u>574.4</u>	555.1	545.6	490.8	563.8	482.1	527.2	↓ 9.2%
COCO	ViLT	441.5	372.2	372.6	362.9	396.7	378.1	<u>432.0</u>	365.4	193.7	281.1	366.1	398.1	422.4	327.1	402.2	229.8	425.8	333.9	356.5	↓ 19.3%
	CLIP ZS	394.5	363.0	361.2	330.2	368.7	358.7	391.6	362.2	294.6	294.7	329.0	371.8	<u>391.9</u>	356.4	369.7	308.2	388.0	314.9	350.3	↓ 11.2%
	CLIP FT	420.5	367.2	365.3	331.7	381.5	371.0	412.2	374.4	291.0	289.3	337.3	389.9	<u>413.9</u>	371.7	379.7	306.4	402.1	310.2	358.5	↓ 14.7%
	TCL ZS	477.2	419.8	418.4	418.4	439.0	400.0	450.8	357.5	177.3	316.5	372.0	400.6	<u>452.2</u>	416.1	369.0	190.3	442.7	280.1	371.8	↓ 22.1%
	TCL FT	497.2	454.3	454.4	453.9	468.1	447.8	<u>491.9</u>	433.8	259.9	408.9	443.2	470.1	489.1	467.8	438.2	309.1	474.9	360.9	430.9	↓ 13.3%
	ALBEF FT	504.6	460.0	460.6	460.3	376.4	447.1	493.0	436.5	282.2	408.8	449.8	472.6	493.8	452.1	455.0	347.0	480.9	475.8	438.3	↓ 13.1%
	BLIP FT	516.6	471.9	472.1	467.7	489.5	466.1	<u>507.2</u>	451.7	291.6	432.8	471.8	494.2	506.8	470.4	472.3	404.7	499.6	402.9	458.7	↓ 11.2%
Dataset	Method	Character-level						Word-level						Sentence-level						ave	MMI
		Clean	Keyboard	OCR	CI	CR	CS	CD	SR	WI	WS	WD	IP	Formal	Casual	Passive	Active	Back_trans			
Flickr30K	ViLT FT	522.0	385.3	461.9	388.0	386.2	395.6	398.6	471.9	492.2	480.1	489.8	507.7	<u>510.1</u>	504.5	488.1	508.3	500.1	460.5	↓ 11.8%	
	CLIP ZS	533.7	431.8	478.2	450.5	435.2	444.6	451.3	497.1	509.6	503.3	514.1	519.4	<u>531.7</u>	529.3	524.8	531.4	524.2	492.3	↓ 7.8%	
	CLIP FT	544.3	458.4	500.1	477.6	461.6	471.1	475.5	515.4	530.4	526.0	531.1	536.4	<u>545.8</u>	542.1	537.9	545.1	537.3	512.0	↓ 5.9%	
	TCL ZS	563.8	433.3	499.9	443.3	428.4	444.4	448.9	511.9	523.8	519.1	528.8	<u>548.6</u>	544.4	542.4	530.1	547.1	535.8	501.9	↓ 11.0%	
	TCL FT	573.4	494.3	545.0	504.9	492.8	501.9	502.4	554.7	566.4	560.0	564.2	<u>573.4</u>	571.5	569.6	562.8	572.1	566.5	543.9	↓ 5.1%	
	ALBEF FT	577.7	506.2	552.0	516.2	505.0	511.7	513.0	561.9	571.6	568.6	570.0	<u>577.7</u>	576.2	575.0	569.5	576.4	572.5	551.5	↓ 4.5%	
	BLIP FT	580.9	518.0	559.5	527.3	518.0	526.4	525.7	565.6	576.1	572.8	573.8	<u>580.7</u>	579.0	578.6	574.5	579.6	574.7	558.1	↓ 3.9%	
COCO	ViLT	441.5	319.2	386.2	327.0	321.7	333.1	334.1	397.8	417.5	404.4	413.6	433.1	<u>436.5</u>	433.6	423.2	437.1	426.0	390.3	↓ 11.6%	
	CLIP ZS	394.5	285.5	286.4	286.1	285.4	285.6	285.8	347.5	363.8	355.5	368.6	374.2	393.0	391.6	379.6	<u>393.5</u>	381.2	341.5	↓ 13.4%	
	CLIP FT	420.5	316.1	316.7	316.5	316.4	316.7	315.6	376.2	394.6	389.9	395.3	406.6	417.3	415.2	408.7	<u>419.4</u>	406.2	370.5	↓ 11.9%	
	TCL ZS	477.2	368.0	428.4	381.3	368.4	382.0	383.4	439.3	453.4	445.7	450.9	<u>477.2</u>	474.4	471.8	464.7	475.7	462.0	432.9	↓ 9.3%	
	TCL FT	497.2	397.8	455.1	412.0	398.5	408.8	410.5	463.7	481.3	471.8	477.7	<u>497.1</u>	494.6	493.0	487.3	496.0	483.5	458.0	↓ 7.9%	
	ALBEF FT	504.6	404.5	461.7	418.9	406.1	414.7	415.5	471.4	488.9	483.3	486.3	<u>504.5</u>	503.1	502.0	496.4	503.7	491.3	465.8	↓ 7.7%	
	BLIP FT	516.6	429.1	479.1	442.4	430.8	441.3	441.4	484.3	502.1	494.6	499.7	<u>515.8</u>	514.4	513.6	508.1	515.4	504.3	482.3	↓ 6.6%	

Given the recent paradigm shift to using generative loss objectives in pre-training multimodal models, e.g., BLIP [241], CoCa [536], SimVLM [491] PaLI [59], Unified-IO [273], OFA [480], we believe this observation could be generalized to other multimodal tasks.

We provide qualitative evidence by visualizing the cross-modal alignment between the image patch and word query using optimal transport [212]. As shown in Figure 6.3, when using GT image-text pair, the retrieval model can accurately locate the image patches given word query. After image perturbations, in particular the ones with high impact like *pixelate* and *zoom blur*, we can clearly see that the model has difficulties finding the correct alignment. However, for the “softest” perturbations like *brightness* and *glass blur*, the model is still able to generate a transport plan (OT coupling matrix) between word and image patch. Similarly, in Figure 6.4 where the text are perturbed, we can see the retrieval model cannot locate the correct word query under *keyboard* and *CR*, but still functions well under *IP* and *formal*. Overall, the visualization of word patch alignments in Figure 6.3 and 6.4 confirm the conclusion drawn from Table 6.5, showing that the alignments are worst for perturbations that lead to highest performance degradation.

Visual reasoning and visual entailment These two tasks are commonly considered to be multimodal classification problems. We present the accuracy results in Tables 6.6. For both the visual reasoning (VR) and visual entailment (VE) task, we observe that *zoom blur* consistently impacts the model performance the most. Character-level perturbations show a stronger influence than word-level and sentence-level perturbations, which conform to the observation for image-text re-

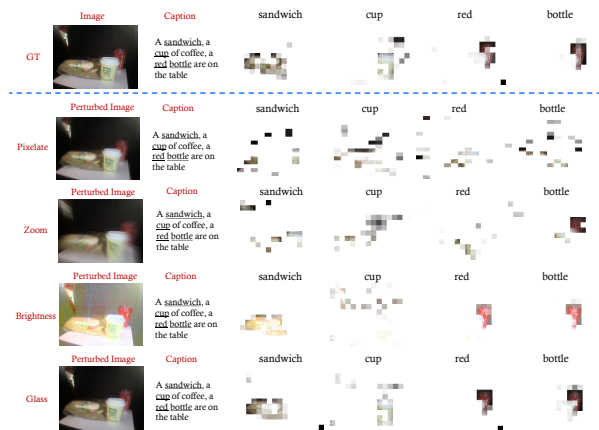


Figure 6.3: OT alignment visualization between text and **perturbed images**, where *pixelate* and *zoom blur* are two high-effective image perturbation methods, *brightness* and *glass blur* are two low-effective ones.

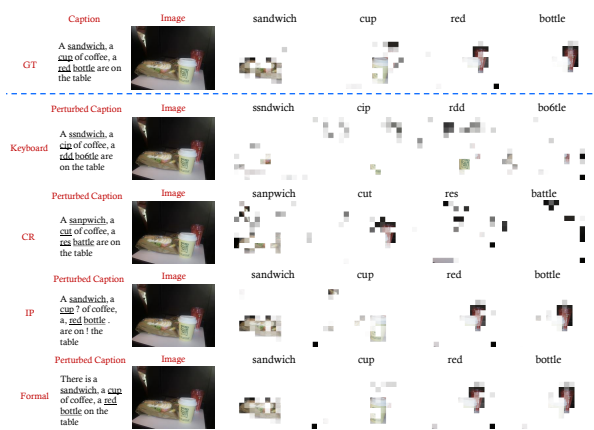


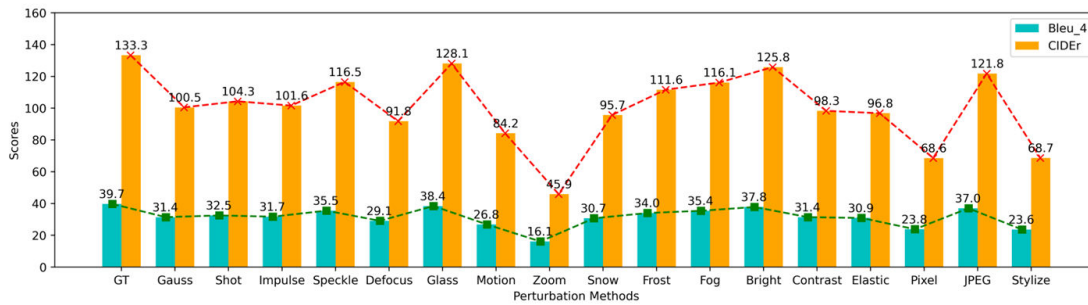
Figure 6.4: OT alignment visualization between **perturbed text** and images, where *keyboard* and *character replace* are two high-effective text perturbation methods, *insert punctuation* and *formal* are two soft ones.

trieval. Note that for visual reasoning, the most influential text perturbations are different across the different models, but they all belong to the character-level perturbation category. *Glass blur* is the “softest” image perturbation for visual reasoning and *brightness* for visual entailment. Regarding text perturbations, *insert punctuation* and sentence-level perturbations like *formal* and *active* have the least impact on the model’s performance for both tasks.

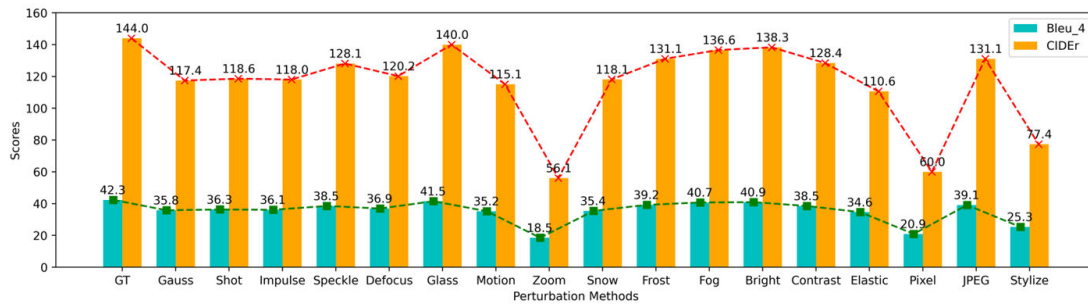
Interestingly, when comparing the robustness of the different models, we make the following observation. Despite TCL is closely related to ALBEF, its robustness performance in terms of MMI score is significantly better. The major difference between both models is that TCL incorporates an intra-modal contrastive loss objective on top of ALBEF, which enforces the learned representations to be semantic meaningful. Additionally to our findings, it has been previously shown that this strategy is also useful in mitigating the noise in training data [523]. Building on these observations, we recommend that we should consider both intra-modal and cross-modal relations in multimodal representation learning to improve the robustness.

Image captioning In this section, we present the image captioning results of BLIP [241] and GRIT [316] under image perturbations. We present the common evaluation metric Bleu_4 and CIDEr in Figure 6.5. As shown in Figure 6.5, *zoom blur* consistently has the most considerable impact across all perturbations on both models. On the other hand, both models are least sensitive to *glass blur*, *brightness*, and *JPEG compression*. In addition, we find that across all considered six evaluation metrics, the CIDEr scores are most sensitive to the perturbations, which suggests it is an informative metric for robustness evaluation.

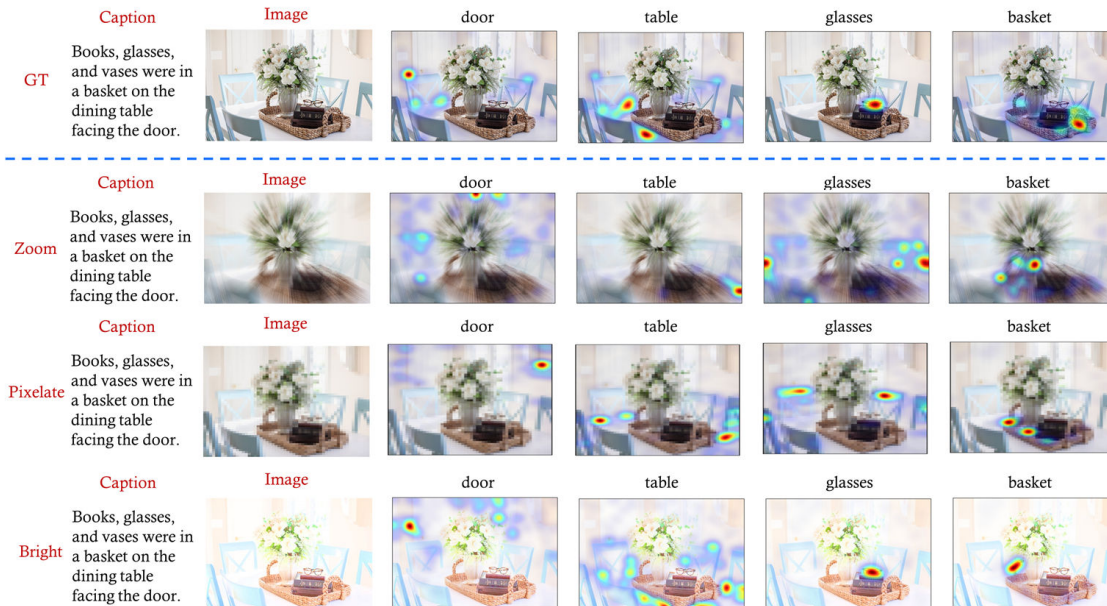
We provide further insights into the effect of the perturbations by inspecting the Grad-CAM [413] visualization of BLIP in Figure 6.5 (c). Given an image, we expect that a robust model is able to attend to different objects according to the word query. Confirming the results shown in the bar plots of Figure 6.5, we find that “hardest” perturbations, including *zoom blur* and *pixelate* distract the attention of the model the most. For instance, BLIP cannot localize the table or the glasses in the perturbed images. However, for “soft” perturbations like *brightness*, BLIP is able to provide reasonable localization.



(a)



(b)



(c)

Figure 6.5: (a) Image captioning results of BLIP; (b) Image captioning results of GRIT; (c) Grad-CAM visualizations on the cross-attention maps corresponding to individual words under image perturbations, where *zoom blur* and *pixelate* perturbed images show worse word-image attention alignment than the *brightness* perturbed image. For example, in *zoom blur* and *pixelate*, the “door” and “glasses” words’ attention maps are not matched with the correct image patches, while in *pixelate*, all words’ attention maps match correctly.

Table 6.6: **Visual reasoning (VR) and visual entailment (VE): [Top]** Robustness evaluations for NLVR2-IP and SNLI-VE-IP datasets. **[Bottom]** Robustness evaluations for NLVR2-TP and SNLI-VE-TP. We report the averaged accuracy where the most effective perturbation results are marked in bold, and the least effective ones are underlined. The MMI impact score is marked in blue, the lower the better.

Dataset	Method	Noise					Blur				Weather				Digital			Stylize		ave	MMI
		Clean	Gauss.	Shot	Impulse	Speckle	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contrast	Elastic	Pixel	JPEG	Stylize		
NLVR2-test	ALBEF	83.14	53.17	52.85	53.22	53.50	52.68	53.09	52.39	51.19	51.60	52.98	<u>53.49</u>	52.78	53.13	53.12	51.72	53.10	52.95	52.76	↓ 36.5%
	ViLT	76.13	74.24	73.80	74.43	74.20	72.32	<u>76.70</u>	72.55	62.34	69.24	73.36	75.05	74.73	68.68	74.07	69.06	76.52	71.50	72.54	↓ 4.7%
	TCL	81.33	78.10	77.87	78.25	78.91	78.00	<u>81.59</u>	78.17	67.81	75.74	79.62	80.64	81.52	74.35	79.76	74.61	81.28	75.85	77.77	↓ 4.4%
	BLIP	83.08	75.39	75.39	85.10	72.31	<u>85.64</u>	79.49	76.92	58.97	80.51	75.90	81.54	76.92	81.03	77.95	73.333	78.97	73.85	77.01	↓ 7.3%
	METER	83.05	78.87	77.94	77.78	79.23	<u>78.97</u>	<u>82.10</u>	79.14	68.89	76.69	80.10	82.25	81.21	78.20	79.91	72.65	80.74	76.93	78.34	↓ 5.7%
SNLI-VE-test	ALBEF	80.91	77.65	77.70	77.40	78.50	76.62	79.25	76.59	71.70	76.31	78.60	78.47	<u>79.77</u>	78.07	78.34	74.42	78.81	74.89	77.24	↓ 8.3%
	TCL	80.29	77.46	77.38	77.30	78.17	76.80	79.27	75.56	71.07	76.13	78.24	78.38	<u>79.19</u>	78.68	77.74	71.76	78.59	74.70	76.85	↓ 4.3%
	METER	81.19	77.16	77.09	76.90	78.58	77.14	80.13	77.39	74.35	77.79	79.84	80.18	<u>80.46</u>	79.18	78.91	72.67	79.32	76.08	77.79	↓ 4.2%

Dataset	Method	Character-level							Word-level					Sentence-level					ave	MMI
		Clean	Keyboard	OCR	CI	CR	CS	CD	SR	WI	WS	WD	IP	Formal	Casual	Passive	Active	Back_trans		
NLVR2-test	ALBEF	83.14	51.39	51.99	51.04	51.26	51.05	51.24	52.69	52.95	52.95	52.88	53.30	<u>53.39</u>	53.06	52.68	53.26	53.23	52.40	↓ 37.0%
	ViLT	76.13	64.85	69.66	66.76	65.64	65.56	65.14	68.96	73.36	71.35	72.53	75.14	75.86	74.27	72.58	<u>77.00</u>	75.70	70.90	↓ 6.9%
	TCL	81.33	71.16	76.31	72.35	71.56	71.90	72.07	75.49	80.03	78.80	78.78	<u>82.88</u>	82.46	81.52	80.25	82.28	81.53	72.37	↓ 11.0%
	BLIP	83.08	67.69	85.64	67.18	67.69	75.90	74.87	69.23	72.82	78.46	83.59	83.59	79.49	<u>87.18</u>	82.05	82.05	74.36	76.99	↓ 7.3%
	METER	83.05	73.10	77.63	74.05	72.49	70.64	74.27	76.10	79.62	75.96	78.55	<u>82.58</u>	81.87	80.42	79.52	82.34	81.45	77.54	↓ 6.6%
SNLI-VE-test	ALBEF	80.91	64.87	71.90	65.99	65.03	66.91	67.27	74.77	74.93	74.90	78.44	<u>80.20</u>	<u>80.20</u>	<u>80.20</u>	<u>80.20</u>	77.31	73.96	↓ 8.6%	
	TCL	80.29	65.27	71.83	65.81	64.66	67.69	67.25	74.59	73.70	74.49	78.01	79.77	79.77	79.77	<u>79.84</u>	79.84	76.62	73.67	↓ 8.2%
	METER	81.19	66.09	74.26	67.39	66.30	68.92	69.71	74.88	73.89	72.95	78.38	76.65	80.96	80.83	<u>81.21</u>	81.05	77.14	74.41	↓ 8.4%

Text-to-image generation We present a robustness evaluation for text-to-image generation using two popular generative models, Stable Diffusion [389] and GLIDE [317], under text perturbations. Due to limited space, we only show results and the analysis for Stable Diffusion here. Since diversity is essential in text-to-image generation, we generate multiple images given one text for a proper analysis. To assess the diversity, we provide three evaluation settings, where each caption in the dataset is used to generate 4, 8, and 16 images. We adopt the common FID [163] score and CLIP-FID [226, 333] score as evaluation metrics and report the mean and standard deviation.

As shown in Figure 6.6 (a) and (b), we surprisingly find that even for the generation task, character-level perturbations affect the robustness of the models the most compared to word-level and sentence-level perturbations. Furthermore, generating more images reduces the variance under each perturbation (e.g., comparing the green against the blue bars). Additionally, we perform a t-test on the generated images and find them to be not correlated after perturbation according to the p-value. This indicates that most text perturbations have an influence on text-to-image generation. Our finding is also corroborated by recent prompt engineering work, where well-designed prompt components can produce coherent outputs [265].

Lastly, we also provide a further inspection of Stable Diffusion by Grad-CAM visualization in Figure 6.6 (c). We use the original unperturbed word query to visualize the attention map. *Keyboard*, *word deletion*, and *casual* are shown as character-level, word-level, and sentence-level perturbation examples, respectively. In *keyboard*, the hydrant is missing; in *word deletion*, the color of the hydrant is incorrect, but no object is missing; in *casual*, the attention map perfectly matches the generated images, which shows character-level perturbations could be more effective than word level and sentence-level perturbations. As the *word deletion* in Figure 6.6 (c), we found Stable Diffusion does not explicitly bind attributes to objects, and the reconstructions from the model often mix up attributes and objects, similar to [373].

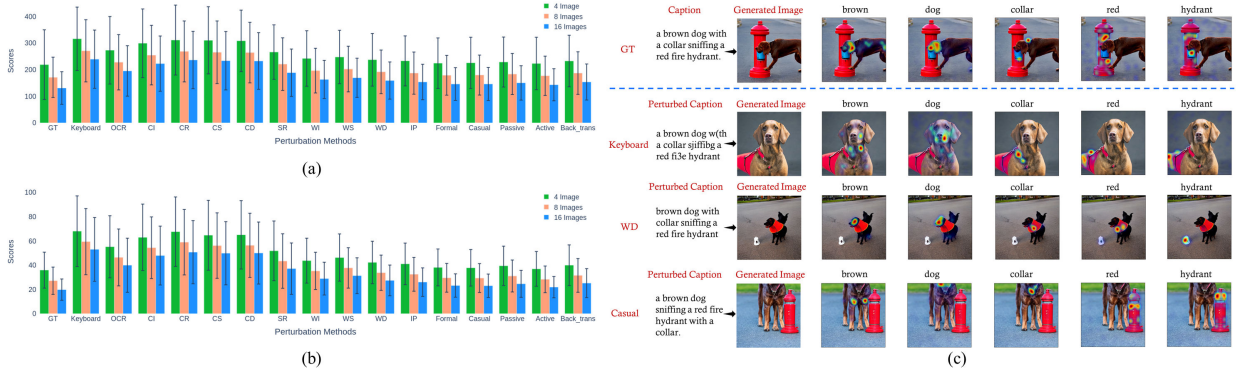


Figure 6.6: (a) Text-to-image generation results of Stable-diffusion in terms of (a) FID scores; (b) CLIP-FID scores. Since both scores are the lower, the better; a higher bar indicates the model is less robust to a particular perturbation. (c) Grad-CAM visualizations on the cross-attention maps corresponding to perturbed captions and images generated by perturbed captions. We use the original unperturbed word query to visualize the attention map. In *keyboard*, the hydrant is missing; in *word deletion*, the color of the hydrant is incorrect, but no object is missing; in *casual*, the attention map perfectly matches the generated images, which shows character-level perturbations could be more effective than word level and sentence-level perturbations.

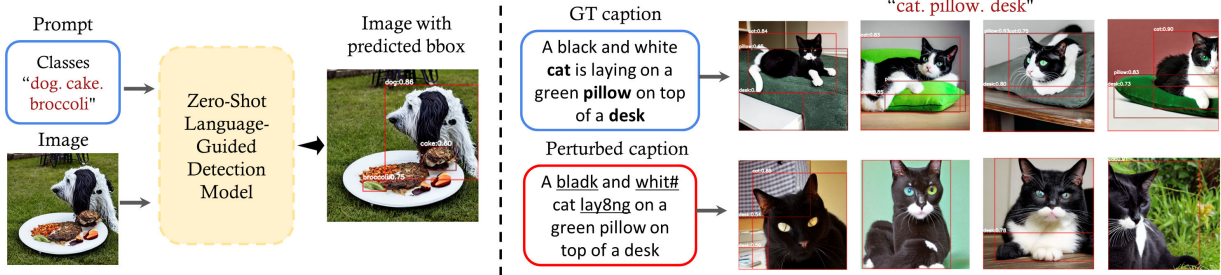


Figure 6.7: Left: Missing Object Rate (MOR) metric calculation. Right: Comparison of detection results between GT-caption-generated images (top) and perturbed-caption-generated images (bottom).

Missing Object Rate (MOR) To further provide a quantitative evaluation of the quality of the generated images, we propose a new detection-based metric to capture if the model can faithfully generate images with all the objects mentioned in the text. To achieve this goal, we leverage an open-set zero-shot language-guided object detection model, i.e., GLIP [244], to detect salient objects in the generated images. As shown in Figure 6.7 left, the inputs to the GLIP model are text prompt and the generated images from text-to-image generation models. Given COCO is an object detection dataset, and it has ground truth labels for the objects, we can simply use the combination of object names from the ground truth labels as the text prompt, i.e., “dog, cake, broccoli”. If the ground truth object can be detected (with a detection threshold α), we assume the object is successfully generated by the text-to-image generation model, otherwise, the object is classified as missing.

In Figure 6.7 right, we show a visual comparison of how perturbed captions can affect the generation quality with respect to missing objects. We first use GT captions and perturbed captions to generate some images, and then perform object detection using GLIP on these images. Note that

Table 6.7: Missing Object Rate (MOR) results of Stable Diffusion. The most effective perturbation results are marked in bold, and the least effective ones are underlined. The results show that more objects are missing from the images generated by character-level perturbed captions.

Threshold	Setting	GT	Keyboard	Ocr	CI	CR	CS	CD	SR	RI	RS	RD	IP	Formal	Casual	Passive	Active	Back_trans
0.7	4-images	0.00	-12.47	-5.22	-8.41	-13.25	-12.15	-12.63	-8.23	-3.14	-7.33	-6.05	-2.81	-2.10	-1.42	-1.36	<u>0.27</u>	-0.86
	8-images	0.00	-11.00	-4.27	-6.62	-11.79	-11.09	-10.76	-6.77	-1.62	-6.59	-4.31	-2.83	0.01	0.69	-0.17	<u>1.34</u>	0.44
	16-images	0.00	-11.53	-4.29	-6.96	-11.72	-11.59	-10.86	-6.88	-1.65	-6.66	-4.48	-2.90	-0.16	0.17	-0.75	<u>0.76</u>	0.48
0.5	4-images	0.00	-5.33	-2.97	-2.96	-6.60	-3.97	-2.45	-1.00	0.72	-1.51	-4.63	-1.88	-0.31	-2.18	<u>2.17</u>	-0.30	0.65
	8-images	0.00	-4.94	-2.28	-1.18	-5.83	-2.48	-1.55	-0.34	1.70	-1.26	-2.72	-1.06	0.17	-1.00	<u>3.41</u>	0.42	1.02
	16-images	0.00	-4.95	-1.76	-1.65	-5.02	-2.01	-2.03	-0.62	1.41	-0.90	-2.50	-0.69	0.50	0.08	<u>3.36</u>	0.26	1.41

for all generated images, we always use the same ground truth COCO object names as text prompts. On the top row, we can find that the prompt "cat, pillow, desk" can be detected successfully, which means they are faithfully generated by the Stable Diffusion model. However, for the bottom row, the perturbed prompt (*CR* in this example), some objects can not be detected and are considered as missing, i.e., pillow and desk.

Hence, similar to mean corruption error (mCE) in [508], we define our detection-based score, termed Missing Object Rate (MOR), as $MOR = (N_P - N_{GT})/N_{GT}$. Here N_P is the number of detected objects from images generated by perturbed captions, and N_{GT} is the number of detected objects from images generated by GT captions. A lower score indicates more objects are missing, which suggests the perturbed text has a high impact on the underlying text-to-image generation model. As shown in Table 6.7, we can see that MOR drops significantly for images generated by character-level perturbed captions compared to word-level and sentence-level methods.

Takeaway: Our main findings are as follows.

- (1) Multimodal image-text models are sensitive to distribution shifts caused by image and text perturbations, especially to shifts in the image space.
- (2) For image perturbations, *zoom blur* consistently shows the highest impact on the model’s robustness across 5 tasks, while *glass blur* and *brightness* are the least harmful ones.
- (3) For text, character-level perturbations have a higher impact than word-level and sentence-level perturbations. In particular, *keyboard* and *character replace* consistently show high impact, while *insert punctuation*, *formal*, and *active* are the three least effective ones across different settings.

6.5 Discussion

Are our findings applicable to unimodal models? Given our findings are consistent on five multimodal vision-language downstream tasks, we further investigate whether our findings still hold for unimodal models under distribution shift. To evaluate whether the findings in our image perturbations of multimodal models are consistent with unimodal vision models, we conducted experiments on multiple unimodal vision models. The top1 classification accuracy is shown in Tables 6.8. In the results, we find that *zoom blur* is still very effective in most models, and *brightness* is the most “soft” image perturbation method, which is consistent with the findings in the multimodal setting.

For image perturbations, we evaluate multiple vision models on ImageNet using the same image

Table 6.8: Top1 classification accuracy of unimodal vision models. The most effective perturbation results are marked in bold and the least effective ones are underlined.

Model/corruption	bright	contrast	defocus	elastic	fog	glass	gauss	impulse	jpeg	motion	pixelate	saturate	shot	snow	spatter	speckle	zoom
deit_base_distilled	<u>0.81</u>	0.81	0.57	0.64	0.79	0.59	0.67	0.66	0.69	0.64	0.66	0.80	0.66	0.67	0.74	0.72	0.54
densenet169	<u>0.72</u>	0.61	0.41	0.45	0.60	0.41	0.42	0.37	0.57	0.41	0.52	0.69	0.41	0.42	0.52	0.47	0.38
eca_nfnet_l0	<u>0.79</u>	0.77	0.47	0.52	0.69	0.50	0.40	0.44	0.65	0.59	0.48	0.78	0.39	0.62	0.72	0.55	0.51
efficientnetv2	<u>0.79</u>	0.72	0.51	0.56	0.62	0.53	0.46	0.49	0.68	0.60	0.60	<u>0.78</u>	0.46	0.62	0.72	0.60	0.54
gmlp_s16_224	0.71	0.72	0.42	0.57	0.66	0.46	0.55	0.53	0.59	0.52	0.58	0.70	0.54	0.34	0.60	0.61	0.39
mixer_b16_224	<u>0.71</u>	<u>0.72</u>	0.31	0.44	0.62	0.35	0.31	0.26	0.44	0.41	0.48	0.63	0.29	0.35	0.50	0.38	0.28
mobilenetv3_large	<u>0.71</u>	0.46	0.35	0.47	0.51	0.38	0.33	0.35	0.56	0.47	0.44	0.68	0.33	0.38	0.55	0.45	0.38
pit_s_224	<u>0.79</u>	0.77	0.50	0.56	0.72	0.51	0.64	0.62	0.67	0.58	0.57	0.77	0.62	0.62	0.70	0.68	0.46
regnety_064	<u>0.75</u>	0.54	0.45	0.51	0.61	0.46	0.43	0.41	0.59	0.48	0.46	0.71	0.40	0.46	0.62	0.48	0.44
resmlp_24_224	<u>0.76</u>	0.73	0.48	0.57	0.61	0.51	0.56	0.54	0.60	0.56	0.54	0.75	0.54	0.56	0.64	0.62	0.46
resnet50d	<u>0.78</u>	<u>0.77</u>	0.44	0.46	0.71	0.47	0.41	0.39	0.63	0.50	0.40	0.76	0.41	0.51	0.63	0.53	0.46
resnext101_32x8d	<u>0.74</u>	0.53	0.48	0.52	0.60	0.47	0.42	0.37	0.61	0.52	0.56	0.69	0.40	0.42	0.57	0.49	0.49
swin_small_patch4	<u>0.81</u>	<u>0.81</u>	0.52	0.57	0.75	0.52	0.63	0.63	0.57	0.61	0.40	0.80	0.62	0.65	0.77	0.70	0.52
vit_small_patch16	0.62	<u>0.73</u>	0.45	0.50	0.67	0.47	0.34	0.31	0.55	0.52	0.58	0.56	0.31	0.26	0.53	0.41	0.38

perturbation techniques in our multimodal setting. Interestingly, similar as in multimodal models, for unimodal vision models, *zoom blur* also has the highest impact on the model performance. For text perturbations, we evaluate several language models on IMDB [279] and MultiNLI [498] datasets, which leads to the same conclusions as for multimodal models: character-level perturbations also have more significant impacts than word-level and sentence-level perturbations. These observations can be corroborated by previous robustness studies on language models [32, 97, 263]. In summary, we find that multimodal models show similar vulnerabilities to image and text perturbations as unimodal models in the corresponding modality.

Limitations and future work Given that our work is one of the early efforts in this direction, there are several promising future work directions and limitations that can be improved. First, we adopt synthetic image and text perturbation strategies in our benchmark. Although the proposed text perturbations mimic realistic shifts, an exciting extension of our work will be to analyze real-world distribution shifts [455, 497]. Second, we select 5 important downstream tasks, but there are more tasks, such as visual question answering and visual grounding, that could be analyzed. In addition, we have introduced the MOR metric to evaluate image generation models, but new evaluation metrics beyond existing ones might be needed for proper robustness evaluation under distribution shifts. Third, our study focuses on evaluating image-text models and highlighting failure points. Building on these insights, it is important to investigate methods that improve robustness. The next natural research direction is to study data augmentation techniques for multimodal models [148], which they have shown to be effective in improving the robustness of unimodal models [157, 160, 497]. Given the fact that both unimodal and multimodal models are sensitive to image *zoom blur* and character-level text perturbations, it might be a good practice to involve these data augmentations during model pre-training. Fourth, all considered multimodal models are learned from web-collected data, which likely contains multiple biases and stereotypes, e.g., w.r.t. gender, race, occupation, etc. This is particularly harmful when using large language models like GPT-3 [43], GPT-4 [322], or text-to-image generation models [400]. An important research direction is to study the robustness and fairness of those models in a unified setting.

6.6 Conclusion

In this chapter, we investigate the robustness of large multimodal image-text models under distribution shifts. Our contributions are:

- We introduce several evaluation benchmarks based on 17 image perturbation and 16 text perturbation strategies.
- We study 5 important downstream tasks, including image-text retrieval, visual reasoning, visual entailment, image captioning, and text-to-image generation, and evaluate 9 popular image-text models.
- [363] has 30 stars, 18 clones, 173 viewers, and 17 citations (as of March 24, 2024).
- We hope that our proposed benchmark is valuable for analyzing the robustness of image-text models and that our findings provide inspiration to develop and deploy more robust models for real-world applications.

Part III

Generalization to Interactive Multimodal Environment

Chapter 7

Language-based Scene Summarization for Embodied Policy Learning

In the previous chapters, we have discussed the robustness of multimodal models, and how to improve the performance by learning cross-domain alignment. Starting this chapter, we explore the generalizability of multimodal models, with two focus areas, one in the interactive environments, and the other one in the healthcare domain.

For interactive environments, robot learning with Large Language models (LLMs) hasn't been fully explored yet. LLMs have shown remarkable success in assisting robot learning tasks, i.e., complex household planning. However, the performance of pretrained LLMs heavily relies on domain-specific templated text data, which may be infeasible in real-world robot learning tasks with image-based observations. Moreover, existing LLMs with text inputs lack the capability to evolve with non-expert interactions with environments.

In this chapter, we introduce a novel learning paradigm that generates robots' executable actions in the form of text, derived solely from visual observations. Our proposed paradigm stands apart from previous works, which utilized either language instructions or a combination of language and visual data as inputs. Moreover, our method does not require oracle text summarization of the scene in the testing time, which makes it more practical for real-world robot learning tasks. Our proposed paradigm consists of two modules: the SUM module, which interprets the environment using visual observations and produces a text summary of the scene, and the APM module, which generates executable action policies based on the natural language descriptions provided by the SUM module. We demonstrate that our proposed method can employ two fine-tuning strategies, including imitation learning (IL) and reinforcement learning approaches, to adapt to the target test tasks effectively. We conduct extensive experiments involving various model selections, environments, and tasks across 7 house layouts in the VirtualHome environment. Our experimental results demonstrate that our method surpasses existing baselines, confirming the effectiveness of this novel learning paradigm. The codebase can be found at https://github.com/Jason-Qiu/Embodied_Policy_Learning.

7.1 Introduction

There has been a surge of interest in building LLMs pretrained on large-scale datasets and exploring LLMs' capability in various downstream tasks. LLMs start from the Transformer model [468] and

are first developed to solve different Natural Language Processing (NLP) applications [43, 86, 269]. Recently, LLMs have also shown great potential for accelerating learning in many other domains by generating learned embeddings as meaningful representations for downstream tasks and encoding transferable knowledge in large pretraining datasets.

In this chapter, we focus on the problem of facilitating robot learning by having a LLM in the loop. The robot generates actions according to its environment observations, which are, in general, sensory information in the format of images, point clouds, or kinematic states. We identify one key challenge in massively deploying LLMs to assist robots is that *LLMs lack the capability to understand such non-text-based environment observations*. To solve this challenge, [252] utilize rule-based perception APIs to transform image-based observations into text formats, which then serve as inputs to the LLM. We instead propose to integrate the multimodal learning paradigm to transform images into texts, which allows more principled and efficient transfer to novel robot learning tasks than rule-based APIs. Another key challenge is *the widely-existing large distribution shifts between the training tasks of large pretrained models and testing tasks in the domain of robot learning*. To close the domain gap, [247] adapt the pretrained LLM to downstream tasks via finetuning with observations converted into text descriptions. In the presence of realistic visual observations, an appropriate method to co-adapt pretrained foundation models for testing tasks in robot learning is still being determined.

To address the above challenges, we propose a new visual-based robot learning paradigm that takes advantage of embedded knowledge in both multimodal models and LLMs. To align different modalities in the visual observations and text-based actions, we consider language as the bridge information. We build a scene-understanding model (SUM) with a pretrained image captioning model to grant the robot the ability to describe the surrounding environment with natural language. We then build an action prediction model (APM) with a LLM to generate execution actions according to the scene caption in the format of natural language. To adapt pretrained models in SUM and APM to downstream robot learning tasks, we propose to finetune the multimodal model in SUM with pre-collected domain-specific image-caption pairs and the language model in APM with corresponding language-action pairs. Besides finetuning with expert demonstrations, we further propose a finetuning paradigm of APM based on the sparse environment feedback to endow APM’s capability to evolve with non-expert data. Our contributions are summarised as follows:

- We introduce a novel robot learning paradigm with LLM in the loop that handles multiple modalities of visual observations and text-based actions in a principled manner. We bridge both modalities with natural language generated by a pretrained multimodal model.
- To adapt to target testing tasks, we propose two fine-tuning strategies, including imitation learning and reinforcement learning approaches. We collect a new expert dataset for imitation learning-based finetuning.
- We test the adaptation performance of multiple models of SUM and APM in seven house layouts in the VirtualHome environment. Our experiments demonstrate that our proposed paradigm shows promising results.

7.2 Related Work

Language Models in Robot Learning Recently, several works have successfully combined LLMs with robot learning by taking advantage of the knowledge learned by LLMs i.e., reasoning

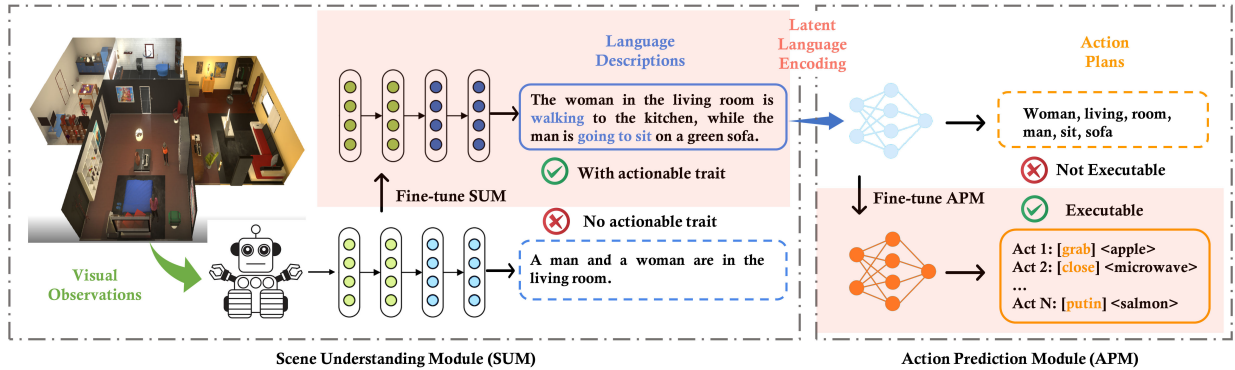


Figure 7.1: The overall architecture of our approach, which includes a scene understanding module (SUM) and an action prediction module (APM). The agent takes pure visual observations and encodes the information as latent language, then the language is transferred to APM for action generation. APM fine-tuned on VirtualHome can generate executable action plans directly.

[252, 543, 544], planning [182, 183, 203, 247, 417], manipulation [45, 196, 209, 383, 416, 421, 422, 448, 515, 516], and navigation [118, 170, 255, 281, 330], which demonstrated the feasibility of using LLM to assist robot learning.

Visual Feedback in Robot Learning Visual feedback is commonly used in robot learning. [131] learned a generative model from actions to image observations of features to control a robot from visual feedback. [278] proposed a self-supervised pretrained visual representation model which is capable of generating dense and smooth reward functions for unseen robotic tasks. [440] reviewed the methods of reward estimation and visual representations used in learning-based approaches for robotics applications. [296] studied the performance of dense, sparse, visually dense, and visually sparse rewards in deep Reinforcement Learning (RL). [225] proposed the direct navigation approach based on an image captioning model. [250] combined image captioning models and planning models, but [250] took pure language instructions as input, while our approach takes pure visual observations as input.

Pre-training and Fine-tuning of Language Models Over the past few years, fine-tuning [174] has superseded the use of feature extraction of pretrained embeddings [337] while pretrained language models are favored over models trained on many tasks due to their increased sample efficiency and performance [392]. The success of these methods has led to the development of even larger models [86, 369]. Fine-tuning pretrained contextual word embedding models to supervised downstream tasks has become commonplace [89, 159]. [544] examined the sampling effects in reinforcement learning with GPT and BERT.

7.3 Proposed Method

In this section, we first introduce our focused problem in Section 7.3.1, which is generating a visual-based policy by leveraging pretrained large models. We then introduce SUM, which learns language descriptions of the surrounding environment in Section 7.3.1, and APM, which predicts actions based on SUM’s caption output in 7.3.2. To grant both SUM and APM the capability of making

the correct understanding and decision in the target domain, we propose finetuning algorithms in Section 7.3.1 and 7.3.2. Our code and data are provided in the supplementary materials.

7.3.1 Problem Formulation

We consider a general and realistic robot learning task where a robot agent receives a sequential visual observation $V = [v_1, v_2, \dots, v_t]$, where t is the timestep, and aims to generate a sequence of actions $A = [a_1, a_2, \dots, a_t]$ based on the pure visual observations V . Traditionally, the robot’s policy is trained from scratch in the target tasks. Inspired by the success of large pretrained models, we aim to explore the benefit of utilizing pretrained LLMs and multimodal models for general robot learning tasks, where only visual observations are available as inputs. Given the prevailing domain shift between the training domain of the pretrained models and the robot learning tasks, we are motivated to develop a principled finetuning method.

SUM: Learning Scene Descriptions from Visual Observations into Language. The goal of the SUM (scene understanding module) is to transform visual observations into language descriptions that contain an actionable trait to it. SUM shares similar functionalities of visual captioning models, which aim to automatically generate fluent and informative language descriptions of an image [206]. For the SUM to be capable of providing scene descriptions from visual observations, it needs to distill representative and meaningful visual representations from an image, then generate coherent and intelligent language descriptions. In our framework, we adopt models with image captioning ability as our SUM, such as OFA [480], BLIP [241], and GRIT [316]. We will discuss the details of possible image captioning models to use in Section 7.4. Generally, image captioning models employ a visual understanding system and a language model capable of generating meaningful and syntactically correct captions [436]. In a standard configuration, the task can be defined as an image-to-sequence problem, where the inputs are pixels, which will be encoded as one or multiple feature vectors in the visual encoding step. The language model will take the information to produce a sequence of words or subwords decoded according to a given vocabulary in a generative way.

With the development of self-attention [468], the visual features achieved remarkable performance due to multimodal pretraining and early-fusion strategies [251, 272, 449, 568]. As for language models, the goal is to predict the probability of a given sequence of words occurring in a sentence. As such, it is a crucial component in image captioning, as it gives the ability to deal with natural language as a stochastic process. Formally, given a sequence of n words y_1, \dots, y_n , the language model component of an image captioning algorithm assigns a probability $P(y_1, y_2, \dots, y_n | \mathbf{X})$ to the sequence as:

$$P(y_1, y_2, \dots, y_n | \mathbf{X}) = \prod_{i=1}^n P(y_i | y_1, y_2, \dots, y_{i-1}, \mathbf{X}) \tag{7.1}$$

where \mathbf{X} represents the visual encoding on which the language model is specifically conditioned. Notably, when predicting the next word given the previous ones, the language model is autoregressive, which means that each predicted word is conditioned on the previous ones. Additionally, the language model usually decides when to stop generating captions by outputting a special end-of-sequence token.

7.3.2 APM: Decoding Language Information into Executable Action Plans

The goal of APM (action prediction module) is to transform latent language information from the SUM output into executable action plans. Since both latent language information and executable action plans are sequential data, a LLM with encoder-decoder architecture is a good option for APM in our framework. In addition, a LLM pretrained on a vast corpus of text already has adequate knowledge, which can be fine-tuned on other tasks to improve learning efficiency.

A LLM with encoder-decoder architecture suits well for our setting. The encoder is responsible for reading and understanding the input language information from SUM, which is usually based on transformer architecture, and creates a fixed-length vector representation, called the context vector. The decoder then takes the context vector as input and generates the output, in our case, the executable action plans. The decoder uses the context vector to guide its generation of the output and make sure it is coherent and consistent with the input information. However, due to the distribution change between the data that LLM was pretrained on and the new task, the LLM needs to be fine-tuned on the task-specific data to transfer the knowledge. The fine-tuning strategies will be introduced in the following sections. For our LLMs, we use well-adopted pretrained architectures, including BERT [86], RoBERTa [269], and BART [235], as both the encoder and decoder. The goal of the LLM is to learn how to generate programmable, executable actions from the language descriptions outputted by SUM.

Algorithm 2 Fine-tuning SUM

```

Initialize pretrained SUM model
Load VirtualHome dataset for fine-tuning
for  $n$  in num_epochs do
  for Image $t$  and Caption $t$  in batch $n$  do
    1.  $\widehat{\text{Caption}}_t = \text{SUM}(\text{Image}_t)$ 
    2.  $\text{Loss}_{XE_t}(\theta_t) = L_{XE}(\text{Caption}_t, \widehat{\text{Caption}}_t)$ 
    3.  $\theta_t \leftarrow \theta_t - \alpha \nabla_{\theta_t} L(\text{Caption}_t, \widehat{\text{Caption}}_t)$ 
  end for
repeat
  Steps 1 through 3
until max(num_epochs) or convergence
end for

```

Algorithm 3 Fine-tuning APM with Imitation Learning

```

Initialize fine-tuned SUM and pretrained APM
Load VirtualHome dataset for fine-tuning
for  $n$  in num_epochs do
  for Image $t$ , Caption $t$ , Action $t$  in batch $n$  do
    1.  $\widehat{\text{Caption}}_t = \text{SUM}(\text{Image}_t)$ 
    2.  $\text{Action}_{t+1} = \text{APM}(\text{Caption}_t, \text{Action}_t)$ 
    3.  $\text{Loss}_{XE_t}(\theta_t) = L_{XE}(\text{Action}_t, \widehat{\text{Caption}}_{t+1})$ 
    4.  $\theta_t \leftarrow \theta_t - \alpha \nabla_{\theta_t} L_{XE}(\text{Action}_t, \widehat{\text{Caption}}_{t+1})$ 
  end for
repeat
  Steps 1 through 3
until max(num_epochs) or convergence
end for

```

7.3.3 Training Pipeline

The training pipeline contains two steps. We first fine-tune SUM with the curated VirtualHome observations (More details about data collection are introduced in Section 7.4.2). This fine-tuning step is to familiarize SUM with the types of scenes present in the task-specific data. We present pseudocode to fine-tune the SUM in Algorithm 2.

In the second stage, we load the fine-tuned SUM and encode the outputs as latent language embeddings. The embeddings are then fed into the APM, which is then fine-tuned using different fine-tuning loss objectives (supervised one or policy gradient, more details are introduced in Section 7.4), to achieve the optimal policy with maximum rewards. The pseudocode for finetuning APM with IL and REINFORCE are in Algorithms 3 and 4, respectively.

Algorithm 4 Fine-tuning APM with REINFORCE

```
Initialize fine-tuned SUM, pretrained APM, and VirtualHome environment (env)
Load VirtualHome dataset for fine-tuning
for  $n$  in num_epochs do
  Trajectories $_t$  = []
  state = env.reset()
  for Image $_t$ , Caption $_t$  Action $_t$  in batch $_n$  do
    1.  $\widehat{\text{Caption}}_t = \text{SUM}(\text{Image}_t)$ 
    2.  $\widehat{\text{Action}}_t = \text{APM}(\widehat{\text{Caption}}_t, \text{Action}_t)$ 
    3. Trajectories $_t.append(\widehat{\text{Action}}_t)$ 
  end for
  sort(Trajectories $_t$ ) by Task ID
  for  $i$  in range(len(Trajectories $_t$ )) do
    4.  $\widehat{\text{Action}}_t = \text{sample\_action}(\text{Trajectories}_t[i])$ 
    5. Reward $_t = \text{env.step}(\text{Action}_t, \widehat{\text{Action}}_t)$ 
    6. Compute  $\nabla_{\theta_t} \log P(\widehat{\text{Action}}_t | \text{Action}_t)$ 
    7.  $\theta_t \leftarrow \theta_t + \alpha r \nabla_{\theta_t} \log P(\widehat{\text{Action}}_t | \text{Action}_t)$ 
  end for
repeat
  Steps 1 through 7
until max(num_epochs) or convergence
end for
```

7.3.4 Fine-tuning APM with IL and RL

For LLM, the output word is sampled from a learned distribution over the vocabulary words. In the most simple scenario, i.e., the greedy decoding mechanism, the word with the highest probability is output. The main drawback of this setting is that possible prediction errors quickly accumulate along the way. To alleviate this drawback, one effective strategy is to use the beam search algorithm [65, 218] that, instead of outputting the word with maximum probability at each time step, maintaining k sequence candidates and finally outputs the most probable one. For the training or fine-tuning strategies, most strategies are based on cross-entropy (CE) loss and masked language model (MLM). But recently, RL-based learning objective has also been explored, which allows optimizing for captioning-specific non-differentiable metrics directly.

Imitation Learning with Cross-Entropy Loss CE loss aims to minimize negative log-likelihood of the current word given the previous ground-truth words at each timestep. Given a sequence of target words $y_{1:T}$, the loss is defined as:

$$L_{XE}(\theta) = - \sum_{i=1}^n \log (P (y_i | y_{1:i-1}, \mathbf{X})) \quad (7.2)$$

where P is the probability distribution induced by LLM, y_i the ground-truth word at time i , $y_{1:i-1}$ indicate the previous ground-truth words, and \mathbf{X} the visual encoding. The cross-entropy loss is designed to operate at the word level and optimize the probability of each word in the ground-truth sequence without considering longer-range dependencies between generated words. Traditional

training with cross-entropy suffers from the exposure bias problem [374] caused by the discrepancy between the training data distribution as opposed to the distribution of its own predicted words.

Reinforcement Learning with REINFORCE Given the limitations of word-level training strategies observed when using limited amounts of data, a significant improvement was achieved by applying the RL approach. Under this setting, the LLM is considered as an agent whose parameters determine a policy. At each time step, the agent executes the policy to choose an action, i.e. the prediction of the next word in the generated sentence. Once the end-of-sequence is reached, the agent receives a reward, and the aim of the training is to optimize the agent parameters to maximize the expected reward [436].

Similar to [374], for our policy gradient method, we use REINFORCE [446, 500], which uses the full trajectory, making it a Monte-Carlo method, to sample episodes to update the policy parameter. For fine-tuning LLMs using RL, we need to frame the problem into an Agent-Environment setting where the agent (policy) can interact with the environment to get the reward for its actions. This reward is then used as feedback to train the model. The mapping of the entities is from the agent (policy), which is an LLM, and the environment (the reward function, also named the model), which generates rewards. The reward function consumes the input as well as the output of the LLM to generate the reward. The reward is then used in a loss function, and the policy is updated. Formally, to compute the loss gradient, beam search and greedy decoding are leveraged as follows:

$$\nabla_{\theta} L(\theta) = -\frac{1}{k} \sum_{i=1}^k ((r(\mathbf{w}^i) - b) \nabla_{\theta} \log P(\mathbf{w}^i)) \quad (7.3)$$

where \mathbf{w}^i is the i -th sentence in the beam or a sampled collection, $r(\cdot)$ is the reward function, and b is the baseline, computed as the reward of the sentence obtained via greedy decoding [385], or as the average reward of the beam candidates [71]. Note that, since it would be difficult for a random policy to improve in an acceptable amount of time, the usual procedure entails pretraining with cross-entropy or masked language model first, and then fine-tuning stage with RL by employing a sequence level metric as the reward. This ensures the initial RL policy is more suitable than the random one.

7.4 Experiments

This section introduces the environment we use in the experiments, the experimental settings, evaluations, and results. We would like to answer the following questions with experiments: (1) Can the proposed paradigm take pure visual observations to generate executable robot actions; (2) What kinds of SUM are able to provide better scene descriptions for robot learning; (3) What kinds of APM show better action decoding ability in generating executable actions; (4) What kinds of fine-tuning strategies show better performance under this setting; (5) Can the model achieve consistent performance across different environments?

7.4.1 Environments and Metrics

Environments We build the experiment environments based on VirtualHome [254, 345], a multi-agent, virtual platform for simulating daily household activities. [345]. [345] provides a dataset of

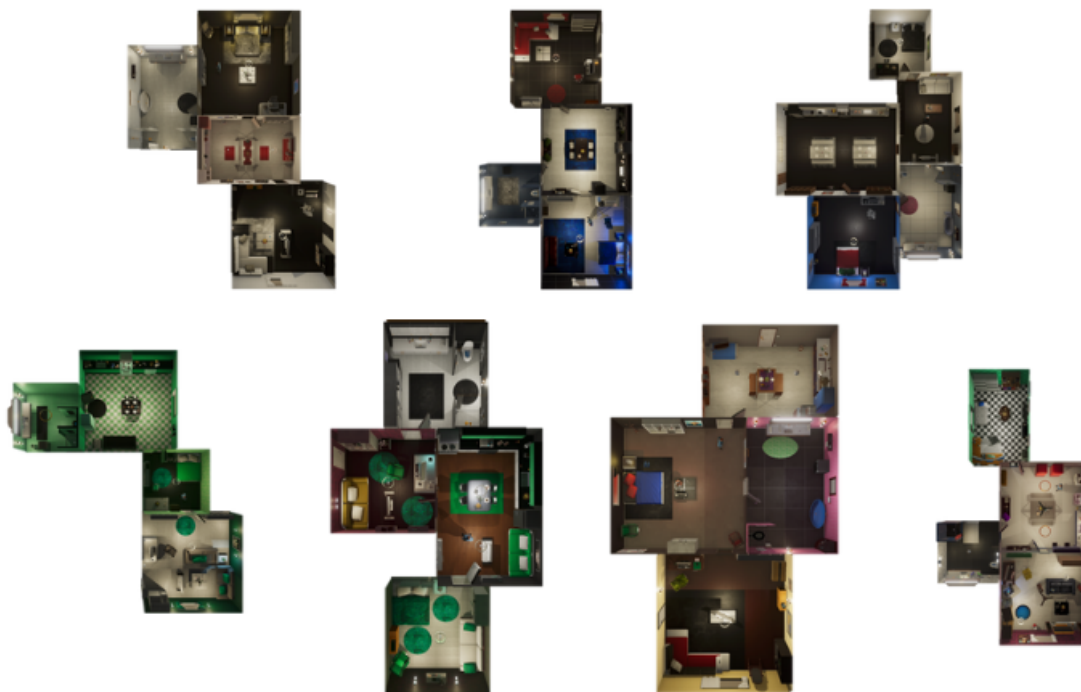


Figure 7.2: Top-down views of 7 different environments from VirtualHome.

possible tasks in their respective environments. Each task includes a natural language description of the task ("Put groceries in the fridge."), an elongated and more detailed natural language description of the task ("I put my groceries into the fridge."), and the executable actions to perform the task in VirtualHome ($[[Walk] < groceries > (1), [Grab] < groceries > (1), \dots [Close] < fridge > (1)]$). We define the training and testing tasks based on the natural language descriptions of the task due to their straightforwardness.

In VirtualHome, the agents are represented as 3D humanoid avatars that interact with given environments through provided, high-level instructions. [345] accumulated a knowledge base of instructions by using human annotators from AMT to first yield verbal descriptions of verbal activities. These descriptions were further translated by AMT annotators into programs utilizing a graphical programming language, thus amassing around 3,000 household activities in 50 different environments [345]. In this study, we evaluate our model’s performance in 7 unique environments, which are shown in Figure 7.2. Each environment has a distinctive set of objects and actions that may be interacted with by agents.

Metrics We use standard NLP evaluation metrics, i.e., BLEU [329], ROUGE [257], METEOR [27], CIDEr [469], and SPICE [15], for evaluating LLMs. In addition, we introduce the execution rate following [247]. The execution rate is defined as the probability of the agent’s success in performing the outputted action from APM over the whole trajectory. We run 10 seeds for each environment.

7.4.2 Datasets

To fine-tune SUM and APM on task-specific robot learning scenarios, we collect data via VirtualHome, including the agent’s observations, language instructions, and action sequences. During

data collection, a household activity program can be described as: $[[action_i] < object_i > (id_i), \dots [action_n] < object_n > (id_n)]$, where i denotes each step of the program, $action_i$ and $object_i$ denotes the action performed on the object at step i , and id_i symbolizes the unique identifier of $object_i$ [345]. The original dataset was augmented by ResActGraph [254]. After augmentation, the dataset contains over 30,000 executable programs, with each environment containing over 300 objects and 4,000 spatial relations. Additionally, we collect the image and text pairs separated by the environments they were executed in. This is important due to the different objects and actions available in each environment. However, as noted in [345] and [254], not all programs were executable.

During data collection, we observe that the text was comprised of two words (e.g., walking bathroom, sitting chair, etc). To have a robust text description, we prompt engineered the texts with a fill-mask pipeline using BERT [86, 429]. For this study, we collect programs executed in three different views: ‘AUTO’, ‘FIRST_PERSON’, and ‘FRONT_PERSON’ as shown in Figure 7.3. In the ‘AUTO’ view, there are locked cameras in every scene through which the program randomly iterates. The ‘FIRST_PERSON’ view observes the agent’s actions through the first-person point of view. The ‘FRONT_PERSON’ view monitors the agent’s actions through the front in a locked third-person point of view. Therefore, the final count of image-text pairs for our dataset in the ‘AUTO’, ‘FIRST_PERSON’, and ‘FRONT_PERSON’ views are 26,600, 26,607, and 26,608, respectively.

More details for data collection For each task, it comprises of a series of actions and there corresponding objects like so: $[[action_i] < object_i > (id_i), \dots [action_n] < object_n > (id_n)]$, where i denotes each step of the task, $action_i$ and $object_i$ denotes the action performed on the object at step i , and id_i symbolizes the unique identifier of $object_i$. For each task, we would simulate it in VirtualHome and output each frame of the task as our visual observations. To conjure up the textual descriptions, we labeled each frame of the task with its corresponding $[action_i] < object_i > (id_i)$ (e.g., walk <bathroom> (1)). We then parsed this into a natural language format (e.g., walk <bathroom> (1) -> walk bathroom).

We notice that the text descriptions were extremely short (e.g., walk bathroom, sitting chair, run treadmill). To create more informative and sensical textual descriptions, we apply prompt engineering by masking in between the action and object (i.e., walking [MASK] bathroom, sitting [MASK] chair, running [MASK] treadmill). This would then give us outputs such as walking **to** bathroom, sitting **on** chair, and running **on** treadmill.

For the finetuning of SUM, we input the visual observation (i.e., the frames gathered during data collection) and output an image caption that serves to describe the scene. We calculate the loss by utilizing the textual description we collected since these textual descriptions are supposed to represent the "action" being partaken during the frame.

Divergence of text-image pairs and the number of the possible agent actions The divergence of text-image pairs is in the different combinations of action and object pairs for each text-image pair. For example, let us say we have a task of “Turn on the Light”, and for this task, there are some actions, such as $[[WALK] < bedroom > (1), [WALK] < lamp > (1), [SWITCHON] < lamp > (1)]$. For each action, there are N images (or frames) and text descriptions. For a given action, each image (frame) is different. However, for the text description, we simply use the same description as the ground truth for describing each image. Nevertheless, there is a diverse corpus of action-object combinations (we have 18 different actions and 308 different objects). In our study, there are

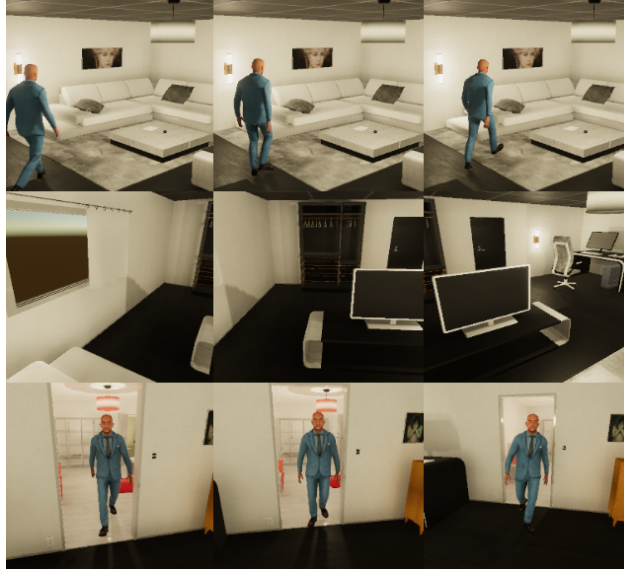


Figure 7.3: ‘AUTO’, ‘FIRST PERSON’, ‘FRONT PERSON’ views.

18 different actions, including [FIND], [TOUCH], [WALK], [SWITCHON], [GRAB], [READ], [STANDUP], [TURNTO], [LOOKAT], [SIT], [POINTAT], [OPEN], [WATCH], [RUN], [DRINK], [SWITCHOFF], [PUTOBJBACK], and [CLOSE].

7.4.3 Experimental Setup

SUM Setting For SUM, we use the following image captioning models to serve as SUM: OFA [480], BLIP [241], and GRIT [316]. Both OFA and BLIP are pretrained on the same five datasets, while the GRIT model [316] is pretrained on a different combination of datasets. For OFA, we adopt OFA_{Large} due to its superior performance in five variations. OFA_{Large} yields ResNet152 [156] modules with 472M parameters and 12 encoders and decoder layers. For BLIP, we use ViT-L/16 as the image encoder due to its better performance. For GRIP, we follow [316] which utilizes the Deformable DETR [579] framework. Note that in our study we want SUM to generate captions that not only describe the scene but also try to derive action from it. We observe that adding the prompt "a picture of " following [491] causes the model to be biased in solely describing the scene, which would in turn not be helpful for generating actionable captions. Therefore, we remove prompts in the SUM setting. We load pretrained models and fine-tune them for 7 epochs on our collected VirtualHome dataset. We keep the hyper-parameters consistent with the original implementations [241, 316, 480].

APM Setting We take LLM to act as our APM. The goal of APM is to generate executable programs for the VirtualHome simulator. We deem the program outputted by the APM executable if the agent in the VirtualHome simulator is able to understand and perform the action. When the action is executed by the agent, the simulator is then directed to output images and captions that are synonymous with the input of SUM. The output of hidden layers of SUM acts as the input embeddings to the APM, while the tokenized executable actions act as labels. The last hidden layer of APM acts as input embeddings for the tokenizer and generates token identifiers. The token identifiers are finally decoded into programmable actions.

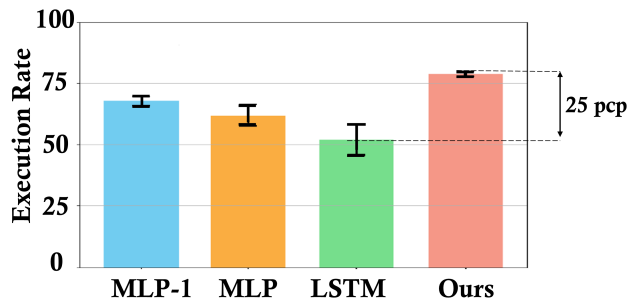


Figure 7.4: Comparison with baselines in the imitation learning setting evaluated by the execution rate.

Training and Testing Tasks . We train and test on seven environments considering that in VirtualHome, there are seven environments in total. We use VirtualHome v0.1.0 due to its stability and to be consistent with previous works. We split the training and testing sets in terms of actions and tasks instead of environments (e.g., 20,000 actions in training and 3,000 in testing; 500 tasks in training, 200 in testing). We do this because each environment has different tasks and actions only executable in the given environment. The boundary between training and testing was chosen randomly based on the distribution of actions and tasks. As mentioned before, if there are a total of 10,000 different tasks or actions, we would randomly split the training and testing set to a proportion of 70:30, respectively. Unseen tasks are defined as tasks that are not included in the training set. For example, if we have the following example task of "Walk to the groceries" (e.g. [WALK] ⟨groceries⟩ (1)) in the training set, we would not have this task in the test set and vice versa.

Executable Actions: Here is the list of all actions executable in VirtualHome: [FIND, TOUCH, WALK, SWITCH ON, GRAB, READ, TURN TO, LOOK AT, SIT, POINT AT, OPEN, WATCH, RUN, DRINK, SWITCH OFF, PUT OBJECT BACK, CLOSE, STAND UP].

7.5 Results and Discussions

7.5.1 Model Performance with IL Fine-tuning

We first want to show the benefit of the proposed framework compared with other model architectures. Concretely, in the IL setting with expert data, we compare the execution rate of our model with the MLP, MLP-1 and LSTM baselines in [247]. Our model has OFA in SUM and BART as APM. Note that all the baselines are not trained by datasets in other domains and have structured text input instead of realistic visual inputs as our proposed model. In the LSTM baseline, the hidden representation from the last timestep, together with the goal and current observation, are used to predict the next action. MLP and MLP-1 both take the goal, histories, and the current observation as input and send them to MLPs to predict actions. MLP-1 has three more average-pooling layers than MLP that average the features of tokens in the goal, history actions, and the current observation, respectively, before sending them to the MLP layer. More details about the baselines can be found in [247]. As shown in Figure 7.4, our approach outperforms baselines in [247] in terms of a higher average execution rate and a smaller standard deviation, though all the methods are trained on expert

Table 7.1: Results by different SUM fine-tuned by imitation learning (IL) objective, where BERT serves as APM. The results are shown on 7 different environments in VirtualHome and also the average performance. The best result in each environment and each SUM model is marked in black and bold. The best SUM result with the highest average performance across 7 environments is marked in orange and bold.

SUM/Results(%)	Environment	Bleu-1	Bleu-2	Bleu-3	Bleu-4	ROUGE-L	METEOR	CIDEr	SPICE	Execution Rate
OFA [480]	1	55.1±0.05	45.4±0.10	36.5±0.20	23.0±0.00	60.0±0.16	33.4±0.00	30.2±0.44	49.9±0.43	78.0±2.39
	2	58.0±0.20	41.7±0.19	35.1±1.01	22.1±0.73	60.1±0.50	34.1±0.52	30.3±0.71	48.1±0.41	79.9±2.37
	3	55.3±0.30	42.3±0.62	34.9±0.15	23.0±0.00	60.5±0.01	34.8±0.64	31.2±0.55	48.4±0.17	80.0±3.29
	4	57.8±0.73	42.2±0.31	35.3±0.38	24.5±0.67	59.9±0.45	34.6±0.54	33.1±0.63	49.0±0.66	79.9±4.14
	5	59.4±0.44	40.3±0.03	34.8±0.02	24.2±0.37	59.7±0.25	35.1±0.62	32.7±0.24	38.0±0.13	77.4±1.12
	6	60.5±0.01	48.1±0.53	36.6±0.07	25.1±0.15	61.9±0.13	36.2±0.60	34.6±1.07	49.9±0.77	80.5±1.13
	7	58.2±0.30	46.5±0.58	34.6±0.04	22.3±0.08	58.3±0.92	35.6±0.62	30.8±0.37	44.2±0.33	69.2±2.31
	Average	57.8±0.92	43.8±1.02	35.4±0.63	23.5±0.77	60.1±0.41	34.8±0.62	31.8±1.31	46.8±0.80	77.8±3.26
BLIP [240]	1	51.1±0.50	42.6±0.41	33.2±0.34	21.1±0.63	60.8±0.73	34.7±0.63	35.5±00.09	42.7±0.91	72.6±1.99
	2	50.5±0.87	41.8±0.72	30.5±28	22.3±0.34	60.3±0.64	33.6±0.87	30.0±0.72	42.8±0.99	66.1±4.21
	3	52.4±0.54	43.2±0.65	33.6±0.13	21.1±0.52	61.4±0.29	34.5±0.12	31.1±0.00	48.9±0.80	85.0±3.32
	4	51.0±1.19	42.1±0.87	33.8±0.54	22.8±0.65	60.6±0.76	34.4±0.98	35.1±0.85	46.0±0.74	73.0±3.65
	5	49.0±0.53	38.8±0.43	30.4±0.72	20.0±0.47	58.6±0.65	34.1±0.75	21.0±0.66	30.8±0.69	67.2±0.93
	6	52.6±0.79	44.5±0.00	31.0±0.63	24.8±0.62	62.0±0.73	35.3±1.02	31.0±0.02	42.4±0.87	84.1±3.54
	7	52.7±0.50	44.0±0.21	33.6±0.18	24.0±0.52	61.7±0.08	34.5±0.60	34.5±0.81	48.8±0.28	86.0±4.92
	Average	51.3±0.31	42.4±0.54	32.3±0.66	22.3±0.31	60.7±0.63	34.4±0.75	31.2±0.87	43.2±0.97	76.3±5.22
GRIT [316]	1	50.5±0.99	40.5±0.86	31.8±1.82	20.7±1.02	60.0±1.44	33.1±0.97	30.4±1.42	41.7±0.85	69.2±5.57
	2	52.1±0.66	41.8±1.77	31.7±1.92	20.1±0.97	59.9±0.65	32.1±0.76	29.4±0.87	42.0±0.88	71.4±5.52
	3	52.3±0.88	40.3±0.82	32.1±0.77	19.9±1.53	60.4±0.68	31.7±0.66	30.1±2.52	43.5±1.64	71.3±5.98
	4	51.9±0.93	39.8±0.92	31.8±0.97	21.3±1.72	59.7±1.22	32.0±0.76	30.0±0.79	42.8±0.84	72.8±4.65
	5	54.7±0.93	42.3±1.02	33.2±1.25	24.5±0.93	62.3±1.42	33.8±1.77	30.7±1.32	44.6±1.23	78.5±5.07
	6	54.6±1.42	44.7±1.64	34.1±1.32	25.8±1.22	65.8±1.25	30.1±2.31	34.5±0.72	44.0±0.96	78.4±3.66
	7	53.9±0.88	42.0±1.79	32.6±2.00	22.5±0.90	63.4±1.00	31.8±1.23	32.3±1.31	43.1±1.41	70.0±3.99
	Average	52.9±0.18	41.6±0.87	32.4±0.72	22.1±0.68	61.6±0.53	32.1±0.33	31.1±0.25	43.1±0.76	73.1±3.11

Table 7.2: Execution Rates by different SUM fine-tuned by REINFORCE, where BERT serves as APM. The results are shown on 7 different environments and also the average performance. The best results are marked in bold.

SUM	Env-1	Env-2	Env-3	Env-4	Env-5	Env-6	Env-7	Average
OFA [480]	50.1±0.65	50.3±0.52	51.5±0.48	57.8±0.88	55.2±0.00	56.6±0.37	59.3±0.48	54.4±0.55
BLIP [240]	52.7±0.78	53.4±1.00	53.5±0.92	55.6±0.68	60.1±0.49	59.3±0.91	49.9±0.90	54.9±1.99
GRIT [316]	38.7±1.02	40.0±1.11	51.3±0.99	48.2±0.90	46.5±0.85	55.8±0.70	45.3±1.08	46.5±2.01

data with imitation learning objectives. The results show that the pretrained embeddings and large model architecture benefit the performance in downstream robot learning tasks.

7.5.2 Model Performance with RL Fine-tuning

We provide the model performance after fine-tuning SUM with a frozen BERT in Table 7.1 for the IL setting with expert data and in Table 7.2 for the RL setting. The results after fine-tuning APM with the fine-tuned SUM are shown in Table 7.3 and Table 7.4. We found that fine-tuning with expert data in IL results in higher average and per-environment performance than fine-tuning with RL, which shows the benefit of having access to the expert datasets. However, fine-tuning with RL still brings performance improvement to 57.2% as in Table 7.4. Note that without finetuning, the outputs of the LLMs in APM are generally not executable as shown in Figure 7.1. Moreover, we consistently observe that the combination of having OFA in SUM and BART as APM achieves the best performance after both IL (Table 7.3) and RL (Table 7.4) fine-tuning.

Table 7.3: Results by different APM fine-tuned by imitation learning (IL) loss objective. The results are shown by the average of 7 different environments in VirtualHome. The best results are marked in bold.

APM	SUM	Bleu-1	Bleu-2	Bleu-3	Bleu-4	ROUGE-L	METEOR	CIDEr	SPICE	Execution Rate
BERT	OFA	57.8 ±0.92	43.8 ±1.02	35.4 ±0.63	23.5 ±0.77	60.1±0.41	34.8 ±0.62	31.8 ±1.31	46.8 ±0.80	77.8 ±3.26
	BLIP	51.3±0.31	42.4±0.54	32.3±0.66	22.3±0.31	60.7±0.63	34.4±0.75	31.2±0.87	43.2±0.97	76.3±5.22
	GRIT	52.9±0.18	41.6±0.87	32.4±0.72	22.1±0.68	61.6 ±0.53	32.1±0.33	31.1±0.25	43.1±0.76	73.1±3.11
RoBERTa	OFA	57.7 ±0.01	43.2 ±0.00	35.6 ±0.48	24.1 ±0.36	59.9±0.26	34.7 ±0.51	31.4±0.47	47.3 ±0.38	75.4±3.86
	BLIP	50.5±0.71	41.1±0.29	32.0±0.11	23.5±0.64	61.1 ±0.88	33.0±0.70	31.8 ±0.81	42.9±0.94	77.7 ±0.71
	GRIT	53.1±1.02	42.0±0.90	34.1±1.01	23.1±1.22	60.4±1.92	31.5±0.59	31.5±1.42	42.8±1.77	75.4±4.39
BART	OFA	59.5 ±0.09	45.9 ±0.31	39.8 ±0.37	28.1 ±0.72	61.3±0.65	37.2 ±0.69	34.4 ±0.78	47.0 ±0.88	79.0 ±1.91
	BLIP	52.9±0.80	44.3±0.52	35.5±0.49	25.3±0.62	62.2±1.12	35.3±1.62	32.0±0.97	44.5±0.88	76.0±1.98
	GRIT	54.2±1.68	43.2±1.85	33.6±1.60	25.3±0.93	62.7 ±1.85	33.8±0.62	33.7±0.74	44.7±1.12	77.9±1.77

Table 7.4: Results by different APM fine-tuned by REINFORCE loss objective, averaging on 7 different environments. The best results are marked in bold.

APM	SUM	Execution Rate (%)
BERT	OFA	54.7 ±1.15
	BLIP	54.1±1.37
	GRIT	53.9±3.00
RoBERTa	OFA	55.6 ±4.31
	BLIP	55.2±1.16
	GRIT	54.8±2.54
BART	OFA	57.2 ±2.43
	BLIP	57.0±3.12
	GRIT	55.8±0.99

7.5.3 Ablation Study

To deeply understand the importance of different components in our paradigm that affect the overall performance, we conduct ablation studies on different factors including different components in SUM, different components in APM, and different environment variations.

Different Components in SUM The performances of using different components in SUM for IL and RL fine-tuning are in Table 7.1 and Table 7.2, respectively. From Table 7.1, we see that with expert data, OFA achieves better results than BLIP and GRIT on the average performance over 7 environments. We conjecture that this may be due to OFA being pretrained on 20M image-text pairs, which is larger than the size of other models’ pretraining data. While under REINFORCE fine-tuning loss, as in Table 7.2, BLIP slightly outperforms OFA in terms of average performance but has around 4 times larger standard deviation than OFA.

How Visual Observations Affect SUM Visual observation quality is vital for SUM. In FIRST PERSON view, which lacks explicit action portrayal, SUM faces challenges in generating high-quality textual descriptions. Complex visual scenarios, like blank walls or cluttered scenes with numerous objects, also impede SUM’s ability to provide informative descriptions matching the action or task at hand.

Different Components in APM The results of using different components in APM for IL and RL fine-tuning are presented in Table 7.3 and Table 7.4, respectively. We found that BART consistently

Table 7.5: Comparison of episode success rate.

Method	In-Distribution Tasks	Novel Tasks
[247]	53.7	27.8
Ours (REINFORCE)	58.4	33.7
Ours (Imitation Learning)	68.4	44.8

outperforms other LLMs in both settings. We hypothesize that due to BART’s architectural nature as a denoising autoencoder, it is more suitable for translating natural language descriptions into executable action programs for the VirtualHome simulator.

Different Environments To test the performance variations under different environments, we conducted the experiments separately for each unique environment. The results are shown in Table 7.1 and Table 7.2, for fine-tuning SUM under IL and RL settings, respectively. Due to image observation variations having the most impact on SUM instead of APM, so we only test the performance of SUM under different environment settings. Through Table 7.1 and Table 7.2, we could find that the variations exist among different environments. Generally, environment 6 seems to have the easiest environmental settings for the model to learn.

Stability To evaluate the stability of different models under different environments, we also calculated the standard deviation (*std*) of the results across different trials. The results are shown in Tables 7.1,7.2,7.3,7.4, which shows that BART as APM and OFA seems to be more stable than the rest of the combinations.

Analysis on the differences by different models and reasons We found that the APM consistently generated high-quality executable actions and tasks based on metric scores. The primary reason for the substantial performance variations among models was the constraints within the environments. Each environment had predefined sets of actions, objects, and tasks. If the model generated items outside of these predefined distributions, the simulator couldn’t execute them. For example, the model might generate a valid action like $[grab] < bottle > (1)$, but if the “bottle” object wasn’t predefined in that environment, the simulator couldn’t execute the action. This environmental constraint led to the observed performance variations.

Fine-tuning performance on in-distribution tasks and unseen tasks To further support our findings, we conducted additional experiments that tested the fine-tuning performance on in-distribution tasks and unseen tasks in the VirtualHome environment following the setting in [247]. [247] used reinforcement learning to adapt to downstream tasks. It’s important to note that [247] used oracle text-based inputs that summarize the current observation, whereas we use raw image inputs and understand the scene with our fine-tuned SUM module. We measure the performance with the episode success rate and summarize the main comparison results with [247]) in Table 7.5. Our results show that when fine-tuning with REINFORCE, our method outperforms [247] in both in-distribution tasks and novel tasks. Additionally, when expert data is available in the downstream tasks, fine-tuning with imitation learning outperforms the REINFORCE approach.

Importance and necessity of fine-tuning To underscore the importance and necessity of fine-tuning, we present additional zero-shot testing performances without fine-tuning in Table 7.7

Table 7.6: Our fine-tuning results for different SUM/APM configurations in in-distribution and novel tasks, as well as using REINFORCE and imitation learning strategies. We measure the performance based on the episode success rate.

SUM	APM	In-Distribution REINFORCE	Novel Tasks REINFORCE	In-Distribution Imitation	Novel Tasks Imitation
OFA	BERT	56.1	31.4	65.2	40.7
	BART	58.4	33.7	68.4	44.8
	RoBERTa	51.7	32.3	66.0	42.8
BLIP	BERT	53.7	28.5	61.1	39.5
	BART	55.2	31.2	64.3	40.3
	RoBERTa	50.6	29.3	62.8	39.8
GRIT	BERT	50.5	28.8	61.3	40.4
	BART	51.2	30.0	63.7	39.6
	RoBERTa	49.0	27.1	59.2	38.7

Table 7.7: Comparison action execution rates in zero-shot and fine-tuned settings using both REINFORCE and Imitation Learning.

Method	APM	SUM	REINFORCE	Imitation Learning
1	Zero-shot	Zero-shot	0.1	0.1
2	Zero-shot	Fine-tuned	14.5	21.4
3	Fine-tuned	Zero-shot	5.8	6.9
4	Fine-tuned	Fine-tuned	57.2	77.8

and Table 7.8. Our findings reveal that the episode success rate and action execution rates are significantly lower without fine-tuning in both methods, which highlights the crucial role that fine-tuning plays in improving performance.

How Visual Observations Affect SUM The quality of visual observations has an important effect on SUM. For a view like FIRST_PERSON, where the camera’s perspective is in first person, we noticed that since this view does not explicitly show the agent performing some actions, it was harder for our SUM model to generate high-quality textual descriptions. Another example is the complexity of the visual observation. For example, we found some images of a blank wall or, on the contrary, a very dense observation with many different objects. For such cases, we found that SUM could not generate informative descriptions that fit the action or task being performed.

Analysis on the differences by different models and reasons During evaluation, we tested our models with 10 different seeds to ensure robustness and reported the mean and standard deviations for all models and environments. As per Table 1, it is important to note that the standard deviations for execution rate across all three models are generally pretty high (i.e., ± 0.93 - ± 5.98). We observed that the executable actions and overall tasks generated by the APM were of high quality (as per the BLEU, ROUGE, METEOR, CIDEr, and SPICE scores). We found that the most significant attribute to the high variations in performances was the environment’s constraints. Each environment has a predefined, finite number of actions, objects, and tasks. Therefore, if our model generated some actions, objects, or tasks that are not within the distribution, the simulator would not be able to execute them. For example, our model would generate a sensical action such as $[grab] < bottle >$ (1) in environment 1. However, in this environment, the bottle object was not predefined, thus preventing the simulator from executing the action. This characteristic led to the high variations

Table 7.8: Comparison episode success rate in zero-shot and fine-tuned settings using both REINFORCE and Imitation Learning.

Method	APM	SUM	REINFORCE	Imitation Learning
1	Zero-shot	Zero-shot	0.7	0.7
2	Zero-shot	Fine-tuned	16.7	19.5
3	Fine-tuned	Zero-shot	7.7	8.7
4	Fine-tuned	Fine-tuned	58.4	76.8

of the performance across models. We acknowledge that this bottleneck is important and hope to consider it in future works.

7.6 Limitations and future directions

- We primarily focused on abstract high-level actions represented by language commands, without taking into account low-level controls such as joint motor control. This omission of the low-level control module may limit the overall effectiveness of the learned policies and their ability to function in complex and dynamic environments. An interesting future direction would be to consider the physical capabilities of embodied agents by learning universal low-level controllers for various morphologies.
- Our study might encounter challenges related to long-tailed actions. In our collected dataset, there are actions that occur infrequently, and the current method may not have effectively learned policies for scenarios involving such actions that rarely appear in the collected dataset. This limitation could constrain the overall effectiveness of the learned policies in real-world situations.
- Given that we fine-tuned the model using a dataset collected in the VirtualHome environment, the generalizability of the learned policies to other platforms might be insufficient due to significant differences between various simulated platforms.
- One interesting future direction is extending our proposed framework to solve generalization tasks in a more data and parameter-efficient manner.

7.7 Conclusion

In this chapter, we introduce a novel robot learning paradigm with LLM in the loop that handles multiple modalities of visual observations and text-based actions in a principled manner. We bridge both modalities with natural language generated by a pretrained multimodal model. Our contributions are:

- Our model contains SUM and APM, where SUM uses image observations as inputs taken by the robot to generate language descriptions of the current scene, and APM predicts the corresponding actions for the next step.
- We tested our method in the VirtualHome under 7 unique environments, and the results demonstrated that our proposed paradigm outperforms baselines in terms of execution rates (25 pcp) and shows strong stability across environments.

Chapter 8

Entity-Centric VQA by Retrieval Augmented Multimodal LLM

In addition to the robotics environment discussed in Chapter 7, VQA is another context where incorporating external knowledge sources can be beneficial for answering real-world questions. Vision-extended LLMs have made significant strides in VQA. Despite these advancements, VLLMs still encounter substantial difficulties in handling queries involving long-tail entities, with a tendency to produce erroneous or hallucinated responses.

In this chapter, we introduce a novel evaluative benchmark named **SnapNTell**, specifically tailored for entity-centric Visual Question Answering (VQA). This task aims to test the models' capabilities in identifying entities and providing detailed, entity-specific knowledge. We have developed the **SnapNTell Dataset**, distinct from traditional VQA datasets: (1) It encompasses a wide range of categorized entities, each represented by images and explicitly named in the answers; (2) It features QA pairs that require extensive knowledge for accurate responses. The dataset is organized into **22** major categories, containing **7,568** unique entities in total. For each entity, we curated 10 illustrative images and crafted 10 knowledge-intensive QA pairs. To address this novel task, we devised a scalable, efficient, and transparent retrieval-augmented multimodal LLM. Our approach markedly outperforms existing methods on the SnapNTell dataset, achieving a **66.5%** improvement in the BELURT score.

8.1 Introduction

Vision-extended LLMs have shown significant advancements, excelling at capturing complex semantics and context-aware attributes needed for intricate tasks. However, their abilities in factual VQA tasks, which demand accurate, concrete answers about real-world entities and phenomena, expose certain limitations. Particularly, torso-to-tail or long-tail entities, which constitute a large proportion of real-world data but appear infrequently in training datasets, pose a challenge. This scarcity in representation often leads to VLLMs resorting to generating plausible but incorrect or imaginative content in their outputs, a problem that manifests as "hallucinations" within the context of model responses. To ensure the confident deployment of VLLMs in practical scenarios, there is an urgent need for dedicated research that not only recognizes but actively strives to tackle and reduce instances of hallucinations, especially in the context of factual queries involving these

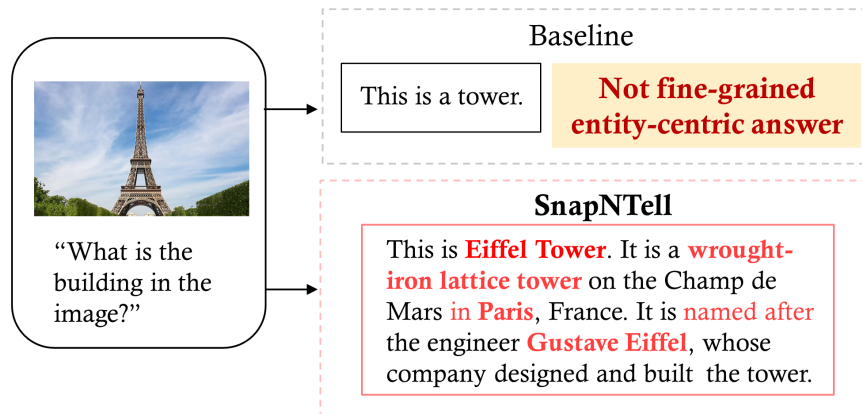


Figure 8.1: Comparing SnapNTell with existing methods reveals a distinctive focus. In the SnapN-Tell benchmark, the answers are predominantly **entity-centric**, characterized by a greater depth of knowledgeable information pertaining to the specific entity depicted in the image as the answer.

long-tail entities.

The lack of publicly available evaluation datasets specifically tailored to assess models’ ability in recognizing real-world long-tailed entities presents a notable gap in VQA. Existing datasets fall short in serving this purpose due to a narrow range of entity categories, the prevalence of overly simplistic yes/no QA pairs, and a general lack of entity specificity, often using broad terms like “Tiger” instead of more specific ones like “Siberian Tiger”. To address this gap, we introduce a novel evaluation task called **SnapNTell**, which focuses on entity-centric knowledge-based VQA. The SnapNTell benchmark has been designed to evaluate models’ abilities in accurately identifying entities and generating responses that showcase a deep understanding of these entities. To support this task, we have curated a new evaluation dataset that departs from existing datasets in two crucial ways: (1) It includes a wide range of fine-grained and categorized entities, each accompanied by corresponding images and clear mention of the entity name within the answer sets. (2) It features QA pairs designed to prompt knowledge-intensive responses, moving beyond the binary yes/no format to challenge and assess the depth of the model’s comprehension.

Furthermore, the limitations identified in factual query generation underscore the need for new solutions to address the problem of hallucinations. Recent advancements suggest that retrieval-based approaches hold significant promise in this regard [142, 435, 525, 528]. These methods enhance LLMs by integrating external knowledge sources or incorporating retrieval mechanisms to access relevant information from extensive knowledge bases. The synergy between the advanced inference capabilities of LLMs and the wealth of external knowledge has the potential to significantly reduce issues related to long-tail entities and, consequently, decrease the occurrence of hallucinatory responses.

In this chapter, we aim to propose an evaluation task to investigate the model’s ability to recognize real-world long-tailed entities and provide knowledge-intensive answers. We also propose a retrieval-augmented method to reduce hallucinations and enhance the precision and trustworthiness of generated responses. Our contribution is summarized as follows:

- **SnapNTell task.** We propose a novel task for **entity-centric** VQA, specifically designed to assess the proficiency of models in accurately identifying and generating responses that exhibit a deep comprehension of these identified entities.

- **SnapNTell model.** We propose a retrieval-augmented multimodal LLM, devised as a baseline model capable of undertaking the SnapNTell task, which is scalable, effective, and explainable.
- **SnapNTell dataset.** We collect a new evaluation dataset with distinctive characteristics, which stands out for two key features: (1) It encompasses a diverse range of fine-grained entities, each accompanied by corresponding representative images. (2) The question-answer pairs contain knowledge-intensive responses with entity names specifically mentioned in the answer sets.
- Our model demonstrates superior performance on the SnapNTell dataset, surpassing current methodologies with a 66.5% improvement in BELURT score.

8.2 Related Works

Knowledge-based VQA Various vision-language tasks often require knowledge to answer questions based on image content and have evolved in recent years. Beginning with datasets like FVQA [479], which extracted facts from pre-established knowledge bases, the field has progressed to more challenging ones like the OK-VQA dataset [285], encompassing diverse knowledge categories. MultiModalQA [447] introduced complexity with questions demanding cross-modal reasoning over snippets, tables, and images. The successor of OK-VQA, AOK-VQA [410], raises the bar by providing questions that transcend simple knowledge base queries. ManyModalQA [147] shifts the focus to answer modality selection, MIMOQA [424] emphasizes multimodal answer extraction, and WebQA [49] introduces real-world knowledge-seeking questions, albeit with some limitations regarding entity categorization and granularity. More comparison details are introduced in Section 8.3.5.

Retrieval augmented LLM Several prior approaches have investigated retrieval-augmented in the text-only setting or image captioning tasks. [142] augmented language model pretraining with a latent knowledge retriever, which allows the model to retrieve and attend over documents from a large corpus such as Wikipedia, used during pretraining, fine-tuning, and inference. [435] demonstrated that retrieval augmentation of queries provides LLMs with valuable additional context, enabling improved understanding. [531] proposed a retriever to retrieve relevant multimodal documents from external memory and use the generator to make predictions for the input. [525] proposed an accelerator to losslessly speed up LLM inference with references through retrieval. [528] introduced a retrieval-augmented visual language model, built upon the Flamingo [8], which supports retrieving the relevant knowledge from the external database for zero and in-context few-shot image captioning. Another related work by [136] integrated implicit and explicit knowledge in an encoder-decoder architecture for jointly reasoning over both knowledge sources during answer generation.

Open-domain visual entity recognition [177] introduced Open-domain Visual Entity Recognition (OVEN) for linking images to Wikipedia entities through text queries. [61] presented INFOSEEK, a Visual Question Answering dataset designed for information-seeking queries. OVEN excels at entity recognition but relies on a knowledge base for entity names, while INFOSEEK primarily provides factual answers. Our research aims to bridge these gaps by generating informative paragraphs that offer context, enabling a deeper understanding beyond mere facts.

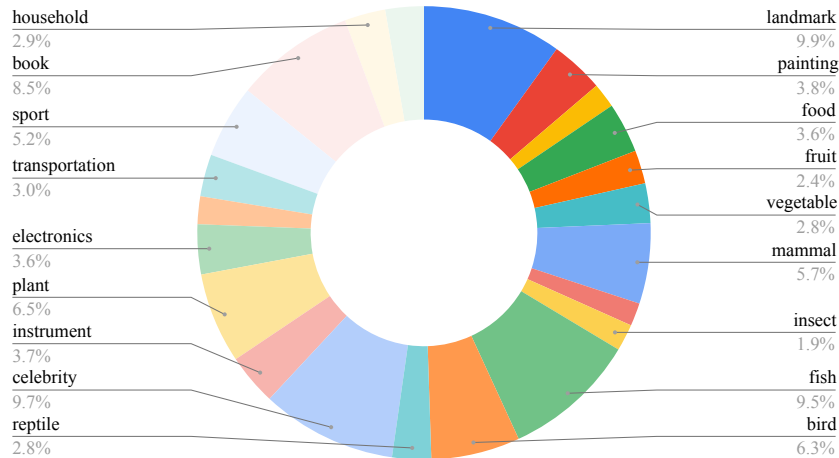


Figure 8.2: Statistics of number of entities in each category.

8.3 SnapNTell Dataset

8.3.1 Entity Categorization

To tackle the challenge of the new SnapNTell task, the first step involves creating a comprehensive dataset that represents a wide array of real-world entities. Our dataset creation methodology entails selecting a diverse set of entity names from various categories that mirror the diversity of the real world. This selection encompasses both commonly encountered entities and less frequently encountered ones. We have identified 22 categories that adequately represent a cross-section of entities one might encounter in daily life. These categories include *landmark*, *painting*, *sculpture*, *food*, *fruit*, *vegetable*, *mammal*, *amphibian*, *insect*, *fish*, *bird*, *reptile*, *celebrity*, *instrument*, *plant*, *electronics*, *tool*, *transportation*, *sport*, *book*, *household*, and *car*. More details about the categories can be referred to Table 8.1.

To populate each category with specific entities, we leverage Wikipedia as a primary resource due to its extensive and detailed entries. Our selection criteria are heavily biased towards specificity; for instance, in the category of mammals, we deliberately opted for precise names such as “German Shepherd” or “Alaskan Malamute” instead of the generic “Dog”. This level of specificity is critical as it enables the model to demonstrate its capacity for fine-grained recognition and its ability to generate detailed, accurate information about each entity. This dataset-building approach is what distinguishes our dataset from existing VQA datasets, which often lack fine-grained entities and specificity.

8.3.2 Image collection

The dataset comprises 22 primary categories, encapsulating a total of 7,568 unique entities. For each individual entity, a set of 10 images has been curated, where the statistic of the entity list is shown in Figure 8.2. The image data collection pipeline is shown in Figure 8.3.

Filtering Initially, a comprehensive list of entities, encompassing 22 primary categories, was compiled, in a total of 14,910 diverse entities. Then the entity list underwent filtering by cross-

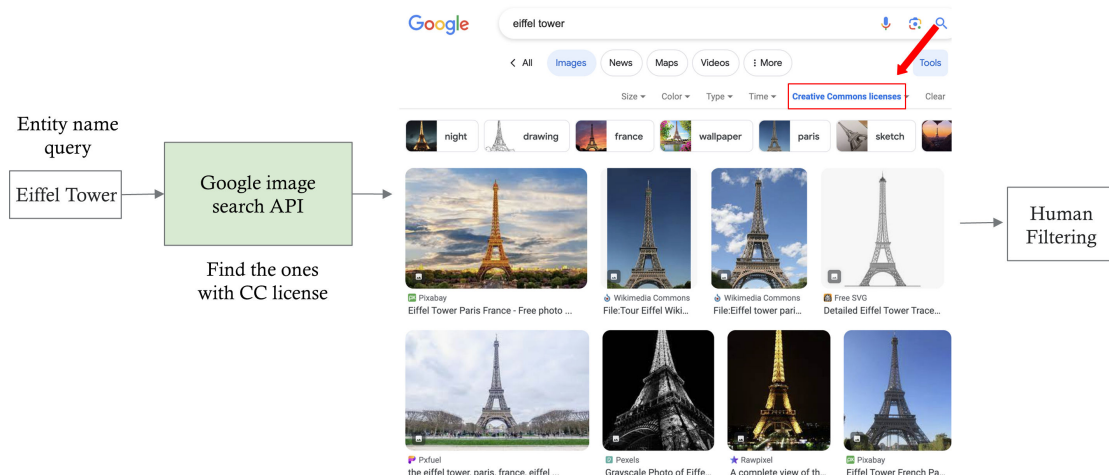


Figure 8.3: Collecting images for building the evaluation dataset. Licenses: CC Publicdomain, CC Attribute, AA Sharealike, CC Noncommercial, or CC Nonderived licenses. Metadata: image URLs, source page URLs, renamed image names, and the corresponding Wikipedia page URL.

referencing each entry with its corresponding Wikipedia page. Entities lacking valid Wikipedia pages were subsequently removed from the list. For each corresponding entity, images were sourced from Creative Commons (CC). Further filtering was conducted by removing entities that didn't have a sufficient number of images obtained via Google Image Search engine. The collected metadata was stored in a CSV file containing essential information such as image URLs, source page URLs, renamed image names, and the corresponding Wikipedia page URLs. After filtering, the final number of entities in the SnapNTell dataset is 7,568. The filtering details are shown in Table 8.1.

8.3.3 Knowledge-intensive Question-Answer Pairs

In our SnapNTell dataset, we consider five types of questions, as shown in Table 8.2. To construct a comprehensive and knowledge-intensive QA dataset, we employ a three-step process. Firstly, we extract and condense pertinent information from Wikipedia for each entity, i.e., the summary of the introduction, the caption of the image, etc. The pipeline is shown in Figure 8.4. Following similar approaches proposed by LLaVA [262], [85] is utilized to generate QA pairs for each entity automatically based on five pre-defined question types, ensuring diversity and informativeness. Then, we enlisted three annotators (2 male and 1 female) from Amazon SageMaker to assess QA pair quality and make necessary revisions to meet specific criteria. The responsibilities of these annotators include: (1) ensuring that the images and QA pairs are semantically aligned, (2) validating the accuracy of the provided answers, (3) making sure the questions are free of particular entity names but demanding such specificity in the answers, (4) assessing if the modified QA pairs adhere to the criteria for knowledge-intensive content, and (5) removing specific entity-related details from the questions. This last step guarantees that the question queries cannot be answered without understanding the accompanying visual context.

Quality and consistency In order to verify the quality of the QA pairs, we conduct a quality evaluation by randomly choosing 1,000 QA pairs from our dataset. We assign three independent human evaluators (1 male, 2 female) from Amazon SageMaker to review these pairs for accuracy

Table 8.1: Filtering statistics of the entity dataset. [1st Wiki filtering]: removing ones without a wiki page. [2nd Google filtering]: removing ones without enough images via google search API. [3rd Wiki filtering]: removing entity names with ambiguous wiki pages.

	Main category	Original Entity	1st Wiki filtering	2nd Google filtering	3rd Wiki filtering
Category	landmark	1595	1000	899	753
	painting	1057	367	358	288
	sculpture	300	164	164	134
	food	883	338	337	271
	fruit	361	236	233	180
	vegetable	389	290	286	214
	mammal	778	633	619	434
	hibian	211	148	139	124
	insect	366	179	176	145
	fish	1089	1054	987	722
	bird	739	546	545	480
	reptile	279	232	231	210
	celebrity	1514	1484	1466	732
	instrument	477	375	368	277
	plant	606	601	593	489
	electronics	432	354	342	269
	tool	801	213	209	150
	transportation	334	296	290	227
	sport	694	478	464	395
	book	1030	826	777	645
	household	475	319	299	221
	car	500	320	320	208
Summary	22	14910	10453	10102	7568

Table 8.2: Types of questions.

Types of questions	Definition
Static facts (absolute facts, discrete facts)	These are objective facts that are concrete and are not contingent on other conditions. They can usually be answered with a short, unique answer. For example: When was Barack Obama born?
Narrative facts	These facts encompass comprehension of larger contexts (e.g., song lyrics, movie plot, historical events). They are factual in the sense that the content of the narrative should accurately reflect the source material or events, but a correct answer is usually not unique, as they can vary in their level of detail and focus. For example: What is the plot of “The Godfather”?
Dynamic facts	These are facts that are subject to change over time. For example: What is the Yelp customer rating of the Eleven Madison Park restaurant in NYC?
Procedural facts	These are usually answers to “how” questions, outlining a sequence of steps to accomplish a task. While the steps may not be unique and could be subjective, in many cases, an answer can still be classified as logical (factual) or nonsensical (a hallucination). Note that these facts can overlap with dynamic facts or narrative facts. For example, How do you check the battery level of my Ray-Ban Stories Glasses?
Subjective facts (opinion-based facts)	These “facts” are not objective, indisputable facts, but are based on individual perspectives or experiences. Recommendations fall in this category. While there’s generally no single correct answer to questions seeking subjective facts, it still requires the system to understand the topic and provide reasonable answers grounded by world facts. For example: Where should I visit Tokyo next month?

Source: Wikipedia



Search Wikipedia Search

Yosemite National Park

Entity: Yosemite National Park

73 languages

Contents hide

(Top)

Toponym

> History

> Geography

> Geology

> Ecology

> Activities

In popular culture

See also

Citations

General references

External links

Knowledge:
General intro

Article Talk

Read Edit View history Tools

From Wikipedia, the free encyclopedia

Coordinates: 37°44′33″N 119°32′15″W﻿ / ﻿37.74250°N 119.53750°W﻿ / 37.74250; -119.53750

"Yosemite" redirects here. For other uses, see Yosemite (disambiguation).

Yosemite National Park (/joʊˈseɪmti/ *yoh-SEM-ih-tee*^[a]) is a national park in California.^[b] It is bordered on the southeast by Sierra National Forest and on the northwest by Stanislaus National Forest. The park is managed by the National Park Service and covers 759,620 acres (1,187 sq mi; 3,074 km²)^[c] in four counties – centered in Tuolumne and Mariposa, extending north and east to Mono and south to Madera. Designated a World Heritage Site in 1984, Yosemite is internationally recognized for its granite cliffs, waterfalls, clear streams, giant sequoia groves, lakes, mountains, meadows, glaciers, and biological diversity.^[d] Almost 95 percent of the park is designated wilderness.^[e] Yosemite is one of the largest and least fragmented habitat blocks in the Sierra Nevada.

Its geology is characterized by granite and remnants of older rock. About 10 million years ago, the Sierra Nevada was uplifted and tilted to form its unique slopes, which increased the steepness of stream and river beds, forming deep, narrow canyons. About one million years ago glaciers formed at higher elevations. They moved downslope, cutting and sculpting the U-shaped Yosemite Valley.^[f]

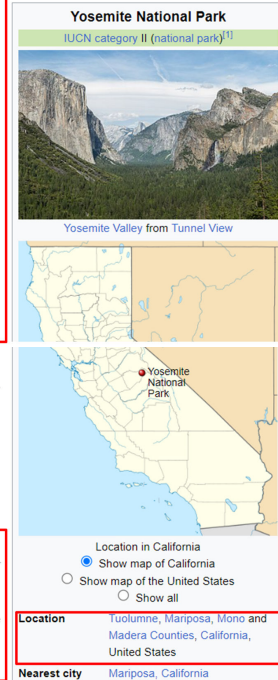
European American settlers first entered the valley in 1851. Other travelers entered earlier, but James D. Savage is credited with discovering the area that became Yosemite National Park.^[g] Native Americans had inhabited the region for nearly 4,000 years, although humans may have first visited as long as 8,000 to 10,000 years ago.^{[h][i][j]}

Yosemite was critical to the development of the concept of national parks. Galen Clark and others lobbied to protect Yosemite Valley from development, ultimately leading to President Abraham Lincoln's signing of the Yosemite Grant of 1864 that declared Yosemite as federally preserved land.^[k] In 1890, John Muir led a successful movement to motivate Congress to establish Yosemite Valley and its surrounding areas as a National Park. This helped pave the way for the National Park System.^[l] Yosemite draws about four million visitors annually.^[m] Most visitors spend the majority of their time in the valley's seven square miles (18 km²).^[n] The park set a visitation record in 2016, surpassing five million visitors for the first time.^[o]

Toponym [edit]

The word *Yosemite* (derived from *yohhe'meti*, "they are killers" in Miwok) historically referred to the name that the Miwok gave to the Ahwahneechee People, the resident indigenous tribe.^{[16][17][18]} Previously, the region had been called "Ahwahnee" ("big mouth") by its only indigenous inhabitants, the Ahwahneechee.^[16] The term *Yosemite* in Miwok is easily confused with a similar term for "grizzly bear", and is still a common misconception.^{[16][19]}

Knowledge:
Toponym



Knowledge:
Location

Figure 8.4: The information collected during dataset building, i.e., from Wikipedia for each entity, which includes the summary of the general introduction, toponym, location information, and so on.

[accurate, inaccurate] and agreement on whether to save the QA pair by Fleiss' Kappa [106]. The outcome of this assessment revealed 98% accuracy and $\kappa = 0.95$ agreement rate among the evaluators, demonstrating a significant degree of uniformity in the quality of the QA pairs.

8.3.4 Statistics and Analysis of Our Dataset

Entity statistics To provide a clear summary of this comprehensive dataset, we have condensed the details of the entity list into Table 8.1 and Figure 8.2. Our analysis indicates that the dataset displays a well-balanced distribution across different categories, enhancing its balanced and diverse characteristics. Such a balanced and diverse composition enhances the representativeness of our proposed evaluation dataset.

Popularity The importance of entity popularity in search engines is a key aspect to consider, similar to examining the head, torso, and tail sections of knowledge bases within search engine frameworks. As demonstrated in Figure 8.5, we use the average Wikipedia pageviews per entity over

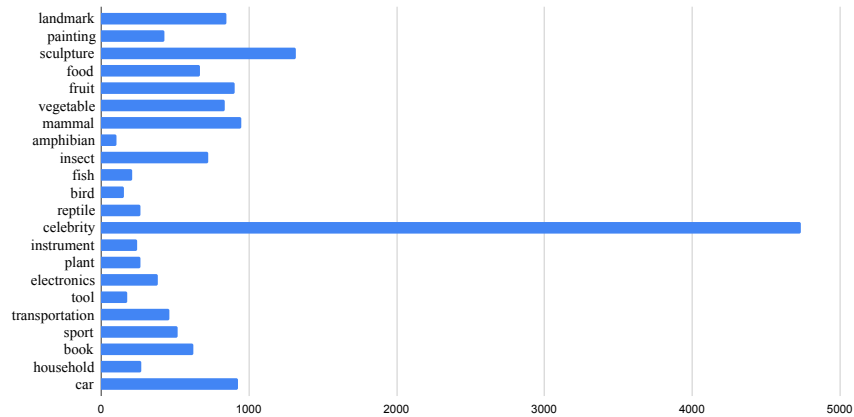


Figure 8.5: Average pageview per entity within each category, where average pageview is defined as the sum of pageviews/ number of entities.

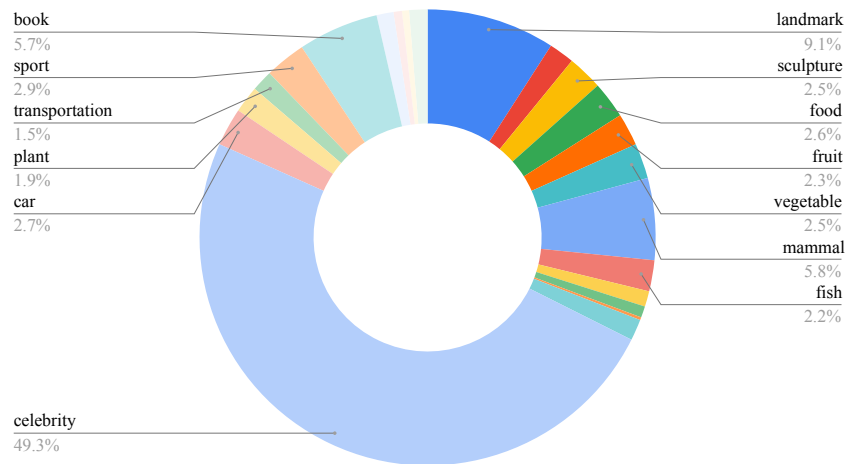


Figure 8.6: Statistics of all pageviews for all categories.

the last 60 days as the metric. This average is calculated by summing up the pageviews and then dividing by the number of entities. The insights from Figure 8.5 reveal that entities in the celebrity category have the highest average popularity. For a broader comparison among different categories, we also present a comprehensive analysis of total pageviews for all categories in Figure 8.6, which shows that the celebrity category remains at the forefront in terms of overall entity popularity. This is attributed to the combination of a higher number of entities in this category and the generally higher popularity of each entity within it.

8.3.5 Comparison with Existing VQA Datasets

In Table 8.3 and Figure 8.7, we present a comparison with existing VQA datasets. It is evident that some existing VQA datasets lack categorization, fine-grained entities, and knowledge-intensive answers, as observed in VQA 2.0 [133] and GQA [188]. OK-VQA [285] contains images that may not be sufficient to answer the questions, encouraging reliance on external knowledge resources.

Table 8.3: Comparison with existing VQA datasets *Knowledge* means the QA pairs are knowledgeable, not simple yes/no answers or selection questions. *Entities* means whether there are fine-grained entities specifically contained in answers. *Categorization* means the entities are categorized, not randomly crawled online.

Dataset	Knowledge	Entities	Categorization
VQA 2.0 [133]			
GQA [188]			
OK-VQA [285]			
ManyModalQA [147]	✓		
MultiModalQA [447]	✓		
MIMOQA [424]	✓		
A-OKVQA [410]	✓		
WebQA [49]	✓	✓	✓
ViQuAE [234]	✓	✓	✓
Encyclopedic VQA [286]	✓	✓	✓
SnapNTell (Ours)	✓	✓	✓

Table 8.4: More detailed comparison with existing knowledge-based VQA datasets. *Anonymity* means whether the question already contains a knowledge clue related to the entity in question. (* Unclear)

Dataset	Categories	Unique Entity	QA Pairs	Images	Average Ans Length	Number of Images / Entity	Anonymity
ViQuAE	3	2,400	3,700	3,300	1.8	*	✗
Encyclopedic VQA (test)	12	*	5,750	5,750	3.2	*	✗
SnapNTell (Ours)	22	7,568	75,680	75,680	25.7	10	✓

However, the answers in OK-VQA are often simplistic binary (yes/no) responses or selections from the questions. A-OKVQA [410], the successor of OK-VQA, aims to provide questions that require commonsense reasoning about the depicted scene but use general object names in the answers. MultiModalQA [447] focuses on cross-modal knowledge extraction but relies on question templates for question generation. ManyModalQA [147] focuses on answer modality choice rather than knowledge aggregation or extraction. In MIMOQA [424], the task of extracting a multimodal answer is not necessarily knowledge-intensive. WebQA [49] does have categorization but lacks fine-grained entities in many QA pairs, resulting in more general questions and answers. Our proposed SnapNTell differs by including a wide range of fine-grained entities with representative images and explicit entity names in the answer sets. Additionally, it incorporates question-answer pairs that demand knowledge-intensive responses, going beyond simplistic binary answers. Examples of our dataset can be found in Figure 8.8.

ViQuAE [234] and Encyclopedic VQA [286] both incorporate entity-level knowledge-based information along with categorization. Therefore, we perform a more in-depth analysis comparing them in Table 8.4. Our dataset surpasses these in terms of the variety of categories, the number of distinct entities, and the overall number of QA pairs. Additionally, our dataset boasts a higher count of images and a longer average length for answers. Specifically, our dataset is structured to include 10 images for each entity, whereas the exact number of images per entity in ViQuAE and Encyclopedic VQA remains unspecified. Most notably, our dataset’s questions are highly anonymous, implying that they do not reveal any knowledge hints about the entity. This design ensures that the questions cannot be straightforwardly answered without interpreting the image data,

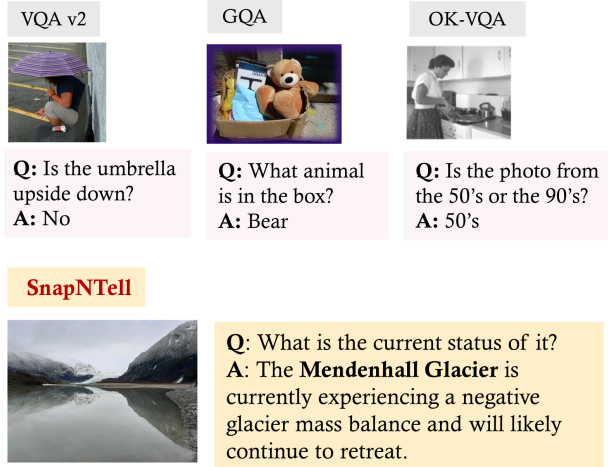


Figure 8.7: Comparison with existing VQA datasets, where previous VQA datasets mostly focus on freeform answers (such as yes/no for verification questions and choice for selection questions).

setting our dataset apart from both ViQuAE and Encyclopedic VQA.

8.4 Proposed Method

In this section, we introduce the details of our proposed retrieval-augmented multimodal LLM model. The architecture of our model is shown in Figure 8.9. Our model can be considered twofold: (1) **Retrieval augmentation**. Given the input image-question pair, we retrieve useful entity-centric information within knowledge sources. (2) **Entity-centric knowledge-based answer generation**. The retrieved information will be combined with the image and question together to generate a knowledgeable answer.

8.4.1 Retrieval Augmentation

The retrieval augmentation process can be subdivided into: (i) Semantic region extraction via language-guided object detection, (ii) Entity recognition via image retrieval, and (iii) Knowledge retrieval via multi-source aggregation.

Semantic Region Extraction via Language-Guided Object Detection To improve recognition performance, we focus on extracting specific image regions containing the entity, rather than general image-level recognition. We employ a language-guided object detection model, i.e., GLIP [244], for language-guided object detection, extracting regions relevant to textual queries by understanding the query context. This targeted approach ensures precise region extraction, enhancing the system’s accuracy and contextual relevance.

Entity Recognition via Image Retrieval To accomplish this goal, we begin by constructing a similarity index using CLIP embeddings, specifically employing Faiss [199] as our indexing tool. Our indexing database is established based on the WIT dataset [434]. This database follows a key-value mapping structure, where the keys represent CLIP ViT-B/32 image embeddings and the




Image	Question	Answer
	Where is it located?	Abel Tasman National Park is located at the northern tip of the South Island of New Zealand between Golden Bay and Tasman Bay
	What date did the it open to the public?	The Acropolis Museum was inaugurated on June 20, 2009, after many years of planning and construction
	What is the architectural style of it ?	The Saint Alexander Nevsky Cathedral has been built in the Neo-Byzantine style.

Figure 8.8: Examples from our SnapNTell dataset.

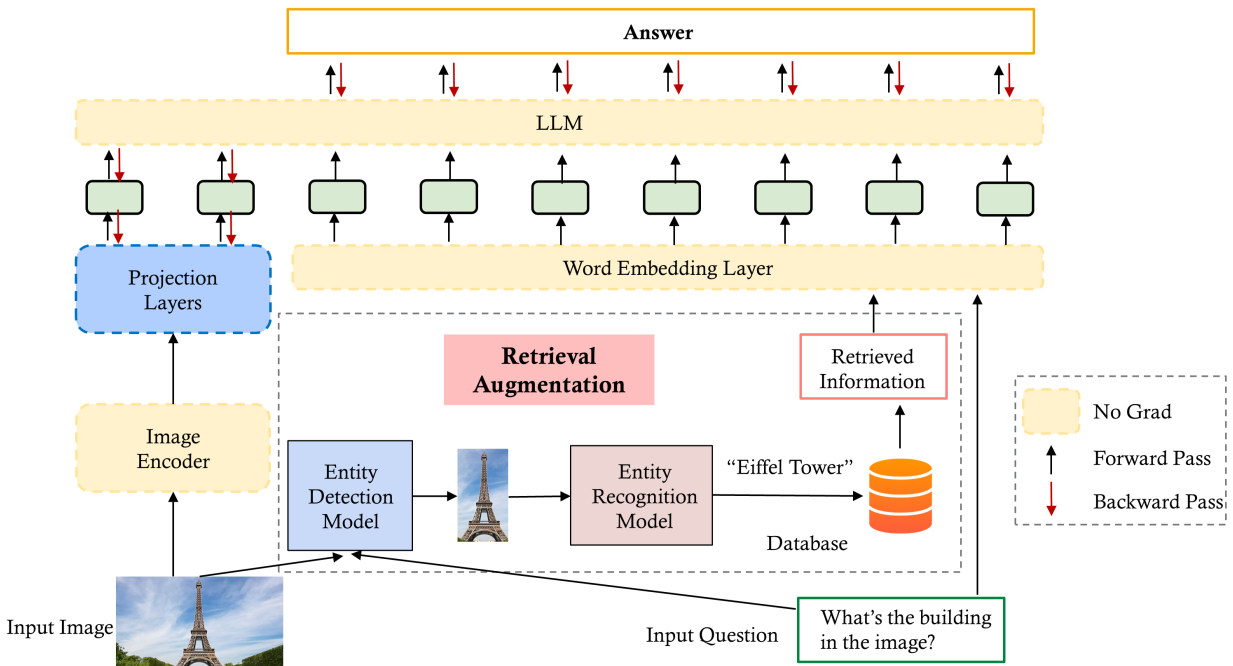


Figure 8.9: SnapNTell model: Our SnapNTell model architecture takes an image-question pair as input. It begins with retrieval augmentation to source relevant information about the entity in the image. This information, along with the question, feeds into the word embedding layer. Text embeddings merge with image-projected embeddings before entering the LLM, culminating in a knowledgeable answer as the output.

corresponding text descriptions serve as the values. Faiss, known for its efficiency in similarity search, is utilized for indexing [199].

After setting up the indexing database, given an input query image I , we perform a k -nearest neighbor retrieval based on cosine similarity. The retrieval outcomes are represented as $\mathcal{R}(I) = \{(i_1, c_1), \dots, (i_k, c_k)\}$, where for each j within the range of 1 to k , i_j and c_j correspond to the retrieved image and its associated caption, respectively. By comparing I with similar images from the database, we identify the entity in the image region, which enables precise image-level entity recognition.

Knowledge Retrieval via Multi-Source Aggregation Facing diverse user queries, we gather extra information to compile resources for accurate responses. Some queries require up-to-date information, not present in existing databases. We then turn to external sources to collect critical data like “year built,” “description,” and more. By using Knowledge Graph (KG) and web searches, we access relevant knowledge links, enriching our understanding of the specified image region, and improving our ability to comprehend and contextualize the extracted content.

8.4.2 Entity-centric Knowledge-based Answer Generation

Following information collection, we enter the integration phase, blending the input image, question, and retrieved data to generate a knowledgeable response, which is illustrated in Figure 8.9. Our method enhances multimodal understanding by pre-training a LLM with image-text paired data. Taking cues from [298], we employ lightweight adapters for each modality, converting inputs into the text token embedding space of the chosen LLM.

Our approach transforms the text token embedding space of the LLM into a unified token embedding space, where tokens can represent either textual or image content. The number of token embeddings allocated to each input modality is predetermined for each adapter, ranging from 64 to 256. Throughout the alignment training process, we keep the model parameters of the underlying LLM frozen. This approach not only accelerates convergence compared to training the model from scratch but also allows the model to inherit the reasoning capabilities of the LLM during inference. Additionally, to maximize feature compatibility, we employ an encoder denoted as $g(\cdot)$ for the image modality. This encoder has previously been aligned with a text embedding space, for instance, in the case of CLIP [367, 406]. For each pair of text and image, represented as $(\mathbf{X}_{\text{text}}, \mathbf{X}_{\text{image}})$, we align them using specific objectives along with a projection module, such as the Perceiver Resampler [8] for the vision encoder.

$$p(\mathbf{X}_{\text{text}}|\mathbf{X}_{\text{image}}) = \prod_{i=1}^L p_{\theta}(\mathbf{X}_{\text{text}}^{[i]}|\mathbf{Z}_{\text{image}}, \mathbf{Z}_{\text{text}}^{[1:i-1]}), \mathbf{Z}_{\text{image}} = \text{Proj}_{\theta}(h_{\text{latents}}, g(\mathbf{X}_{\text{image}})) \quad (8.1)$$

8.5 Experiments and Results

8.5.1 Experimental Setup

Evaluation Metrics (1) In our evaluation process, the quality of the answers is first assessed using established NLP metrics such as BLEU [329], METEOR [84], ROUGE [257], and BLEURT [344, 412]. (2) Additionally, we incorporate accuracy and hallucination rate metrics from [443].

These metrics used [GPT4](#) to automatically measure the proportion of questions for which the model provides correct answers or incorrect/partially incorrect answers, respectively. (3) We conduct human evaluation following [\[298, 532\]](#).

Model Setting We choose LLaMA2 (70B) [\[462\]](#) as our LLM. For image encoding, the CLIP image encoder (ViT-B/32) is employed [\[367, 406\]](#). Additional configurations comprise a batch size of 2,048, the integration of two resampler layers, and the use of 64 modality tokens.

Model Training We use a cleaned subset of the LAION-2B dataset, filtered using the CAT method [\[366\]](#) and with any detectable faces blurred [\[365\]](#). Significant resources are essential to scale pre-training to 70 billion parameter models on a substantial dataset of over 200 million instances. Often, this necessitates the utilization of an FSDP wrapper, as outlined in [\[85\]](#), to distribute the model across multiple GPUs efficiently. To optimize our training process, we employ quantization strategies, specifically 4-bit and 8-bit quantization techniques [\[85\]](#), within our multimodal framework. In this approach, we maintain the LLM component of our model in a frozen state, allowing only the image modality tokenizers to be trainable. This strategy drastically reduces the memory requirements by an order of magnitude. As a result of these optimizations, we can successfully train a 70 billion parameter model on a single GPU with 80GB VRAM, using a batch size of 4.

8.5.2 Results and Discussion

Table [8.5](#) displays the comparative results between the baseline models and our proposed method. Analysis of this table indicates that for every metric assessed, our retrieval-augmented multimodal LLM surpasses the performance of all existing baseline models. This strong performance emphasizes the efficiency of retrieval augmentation in producing responses enriched with entity-centric information, thereby illustrating its substantial impact on the task at hand.

Table 8.5: Performance comparison of different approaches on the SnapNTell dataset.

Method	ROUGE \uparrow	BLEU \uparrow	METEOR \uparrow	BLEURT \uparrow
Instruct-BLIP [77]	10.72	0.95	7.59	0.09
BLIP2 [239]	15.00	0.52	8.49	0.16
Mini-GPT4 [572]	26.12	5.62	25.55	0.27
LLaVA [262]	26.86	6.03	26.97	0.31
Open-Flamingo [21]	30.57	6.52	22.53	0.32
COGVLN [483]	30.25	6.67	23.35	0.31
mPLUG-Owl2 [532]	31.39	6.72	24.67	0.33
LLaVA 1.5 [261]	32.87	6.94	25.23	0.33
SnapNTell (ours)	35.28	7.81	29.27	0.55

Moreover, to gain deeper insights into which evaluation metric more accurately reflects the outcomes, we compute the Kendall correlation coefficient [\[207, 208, 216\]](#), comparing the results with those from the human evaluation in Section [8.5.4](#). Kendall’s τ is a measure of the correspondence between two rankings. Values close to 1 indicate strong agreement, values close to -1 indicate strong disagreement. Table [8.6](#) reveals that both the ROUGE and BLEURT scores are more indicative in distinguishing the differences among various models. This finding suggests that these two metrics

Table 8.6: Effectiveness of evaluation metrics.

	ROUGE	BLEU	METEOR	BELURT
τ	0.999	0.799	0.600	0.999
P_value	0.014	0.050	0.142	0.014

are particularly significant in evaluating model performance in a way that aligns closely with human judgment.

8.5.3 Ablation Study

For a more in-depth understanding, we conduct several ablation studies to delve into the finer details of our approach.

Effectiveness of Entity Detection To assess the impact of entity detection (ED) in our model, we perform an ablation study. This involved comparing the performance of our approach with and without the ED component. As indicated in Table 8.7, our approach incorporating entity detection markedly surpasses the variant lacking this feature. This highlights the significant contribution and necessity of the entity detection step in our model’s overall effectiveness.

Table 8.7: Ablation study on the effectiveness of entity detection (ED).

Method	ROUGE \uparrow	BLEU \uparrow	METEOR \uparrow	BELURT \uparrow
w/o ED	28.02	3.73	26.26	0.45
w/ ED	35.28	7.81	29.27	0.55

Head/Torso/Tail Entities Head knowledge pertains to well-established entities for which there is a wealth of available training data. Ideally, LLMs could be trained to possess this knowledge, facilitating efficient retrieval. On the other hand, torso-to-tail knowledge pertains to less-known or obscure entities, often characterized by scarce or non-existent training data. Providing access to such knowledge involves effectively determining when external information is necessary, retrieving the relevant knowledge efficiently, and seamlessly integrating it into responses.

To assess the performance improvement for head/torso/tail entities, we randomly select 10% entities for each category, where head/torso/tail entities are defined based on pageview statistics (popularity) in Section 8.3.4. The results presented in Table 8.8 clearly demonstrate that retrieval augmentation can significantly enhance performance across various entity types. Notably, the performance improvement for torso-to-tail entities far exceeds that of head entities, effectively addressing the challenge of hallucinations in long-tailed entities through retrieval augmentation.

Performance of Different VQA Datasets To demonstrate the uniqueness of our SnapNTell dataset compared to existing VQA datasets, we analyze the performance of various baseline models on both traditional VQA datasets and our SnapNTell dataset. According to the findings presented in Table 8.9, the performance disparities among baseline models on existing datasets are not particularly marked. In contrast, on the SnapNTell dataset, we observe significantly larger differences and

Table 8.8: Ablation study on head/torso/tail entities, where RA is short for Retrieval Augmentation and Δ is the performance difference of with and without RA.

		Accuracy \uparrow	Hallucination \downarrow
Head	w/o RA	24.4	75.6
	w/ RA	27.1	72.9
	Δ (100%)	11.1 % \uparrow	3.6 % \downarrow
Torso	w/o RA	19.1	80.9
	w/ RA	22.7	77.3
	Δ (100%)	18.8 % \uparrow	4.4 % \downarrow
Tail	w/o RA	6.8	93.2
	w/ RA	12.6	87.4
	Δ (100%)	85.3 % \uparrow	6.2 % \downarrow

notably lower performance. This indicates that our SnapNTell dataset is particularly effective in evaluating the capabilities of different models to recognize entities and produce responses centered around these entities.

Table 8.9: Ablation on the accuracy performance of different VQA datasets (A lower result means the task is more complicated to solve).

Method	VQAv2	TextVQA	OK-VQA	SnapNTell
Instruct-BLIP [77]	–	46.6	55.5	8.88
BLIP2 [239]	52.6	43.1	54.7	16.16
Flamingo [8]	56.3	37.9	57.8	32.17

8.5.4 Human Evaluation Results

In alignment with the methodology presented in [298, 532], we involve a human evaluation process conducted by a panel of five human judges (3 male, 2 female). These judges were given specific instructions for their assessment, which encompasses three key aspects: (1) Recognition Accuracy, where they evaluated whether the model correctly identified the entity in the image relevant to the question; (2) Response Accuracy, in which they assessed the factual correctness of the model’s responses while checking for any signs of hallucination [377]; and (3) Pairwise Comparison, where judges selected the response that better addressed the given question in terms of contextual appropriateness and accuracy, categorizing responses as winning, tying, or losing.

In our study, we conduct pairwise comparisons for each baseline model against ground-truth data across 1,000 samples. As depicted in Figure 8.10, our model outperforms the baselines by displaying a significantly smaller difference when measured against manually annotated ground-truth samples, highlighting its robustness.

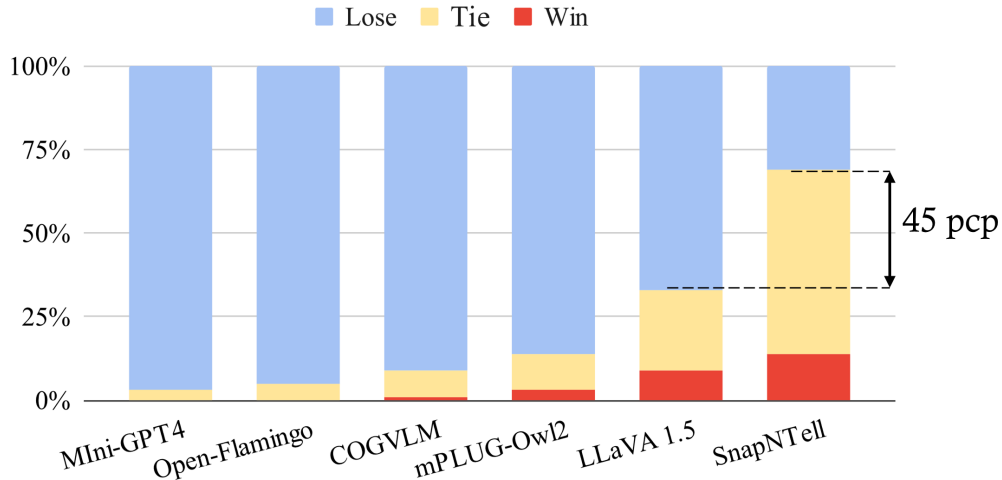


Figure 8.10: Human evaluation results on pairwise comparisons (% win, tie, lose) with baseline outputs *against* the manually annotated ground-truth from SnapNTell.

8.5.5 Discussions

Limitations In this study, we introduce a novel SnapNTell task and its accompanying dataset, which features five unique types of questions, each paired with meticulously formulated answers. It’s important to recognize that in cases involving human preferences, which are subjective by nature, the given answers might not represent the only correct options. Furthermore, the relevancy of some answers may diminish over time, highlighting the need for periodic updates to the dataset to ensure its ongoing relevance and accuracy. Our proposed method exhibited superior performance over existing baselines. However, human evaluation results suggest significant potential for further improvement. Although our approach often neared human-level performance, it did not consistently outperform human annotations, showing opportunities for future advancements.

Broader Impact Current models have made commendable progress in grasping the nuanced semantics and context-sensitive aspects of Visual Question Answering (VQA). However, their efficacy in factual VQA tasks, which require precise and factual answers about tangible entities and events, reveals certain deficiencies. This is especially true for torso-to-tail or long-tail entities. Despite their prevalence in the real world, these entities are underrepresented in training datasets, leading to a common issue where models produce plausible yet inaccurate or invented responses, a phenomenon often termed “hallucinations” in the realm of model-generated content. Tackling and minimizing these hallucinations is vital for enhancing the trustworthiness and applicability of these models in practical scenarios.

The existing VQA datasets, however, are inadequate for evaluating a model’s ability to recognize entities, as they do not explicitly highlight these entities within the dataset. Our newly introduced dataset bridges this gap. It is designed to test models’ capabilities not just in identifying entities but also in generating informed and entity-aware responses. Furthermore, our proposed dataset might serve as resources for either pre-training or fine-tuning existing models, to improve their ability in recognizing entity-level real-world objects.

8.6 Conclusion

In this chapter, we tackle the significant challenge VLLMs face with long-tail entity queries, which often lead to inaccurate or hallucinated responses.

- To address these issues, we introduce an entity-centric VQA task named SnapNTell. This task is designed to test models on entity recognition and their ability to provide detailed, entity-specific knowledge in their responses.
- We collect a unique evaluation dataset for this task, which distinguishes itself from existing VQA datasets by including a wide array of fine-grained categorized entities, supported by images and explicit entity mentions in the answers. This dataset emphasizes knowledge-intensive responses over simple binary answers.
- In addition, we propose a retrieval-augmented multimodal LLM solution for the SnapNTell task as an effective baseline. Our experimental results show that our model outperforms existing approaches, providing more accurate and coherent answers.
- Our approach markedly outperforms existing methods on the SnapNTell dataset, achieving a **66.5%** improvement in the BELURT score.

Part IV

Cross-domain Applications in Healthcare

Chapter 9

Detect Cardiovascular Disease Through Language Models

In Chapters 7,8, we have discussed the generalization capabilities of multimodal models in interactive environments. Recent advancements have drawn increasing attention since the learned embeddings pretrained on large-scale datasets have shown powerful ability in various downstream applications. However, whether the learned knowledge can be transferred to clinical cardiology remains unknown. In the following chapters, we explore the generalization capabilities in the healthcare domain.

In this chapter, we aim to bridge this gap by transferring the knowledge of LLMs to clinical Electrocardiography (ECG). To address this problem, we propose an approach for cardiovascular disease diagnosis and automatic ECG diagnosis report generation. We also introduce an additional loss function by OT to align the distribution between ECG and language embeddings. The learned embeddings are evaluated on two downstream tasks: (1) automatic ECG diagnosis report generation, and (2) zero-shot cardiovascular disease detection. Our approach is able to generate high-quality cardiac diagnosis reports and also achieves competitive zero-shot classification performance even compared with supervised baselines, which proves the feasibility of transferring knowledge from LLMs to the cardiac domain.

9.1 Introduction

Heart and cardiovascular diseases are the leading global cause of death, with 80% of cardiovascular disease-related deaths due to heart attacks and strokes. The clinical 12-lead ECG, when correctly interpreted, is the primary tool to detect cardiac abnormalities and heart-related issues. ECG provides unique information about the structure and electrical activity of the heart and systemic conditions through changes in the timing and morphology of the recorded waveforms. Achievements of ECG interpretation, such that critical and timely ECG interpretations of cardiac conditions, will lead to efficient and cost-effective intervention.

LLM starts from the Transformer model [468] and grows quickly with a wide range of applications [43, 86, 269]. Recently, LLM has shown great potential for accelerating learning in many other domains since the learned embeddings can provide meaningful representation for downstream tasks. Examples include transferring the knowledge of LLM to, i.e., robotics control [7, 252], multimodal

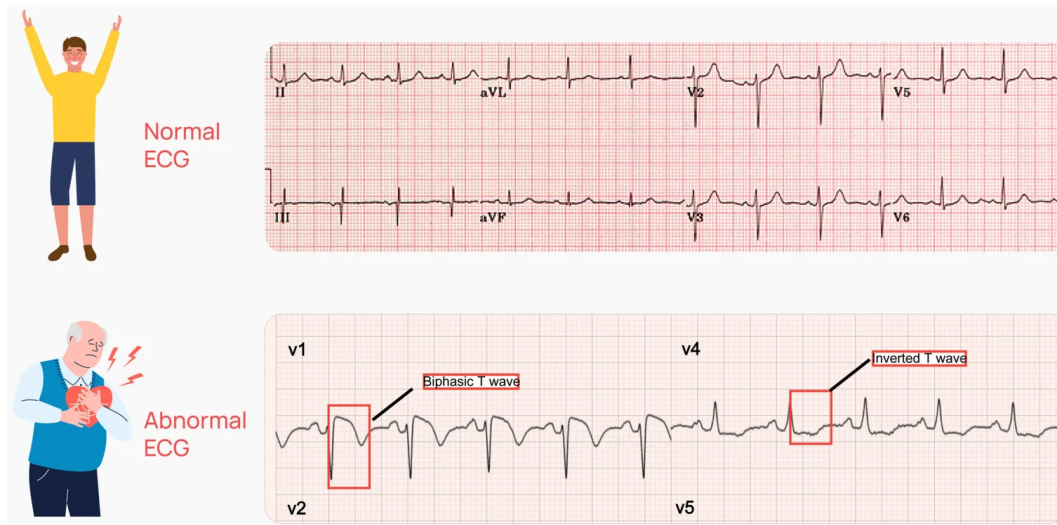


Figure 9.1: Example of 12-lead ECG [444].

reasoning and interaction [543, 544], robotics planning [192, 203, 417], decision-making [182, 247], robotics manipulation [74, 209, 383, 422, 448], code generation [112], laws [204], computer vision [367], and so on.

Some previous works explored LLM and biological protein [387], or health records [526]. However, the medical or healthcare domains contain so much domain knowledge that different sources preserve unique data characteristics without a unified paradigm. To the best of our knowledge, no previous work explores the knowledge transfer from LLM to cardiovascular disease with ECG signals.

In this chapter, we bridge the gap between LLM and clinical ECG by investigating the feasibility of transferring knowledge of LLM to the cardiology domain. Our contributions are listed as follows:

- To the best of our knowledge, our work is the first attempt to bridge the gap between LLM and clinical cardiovascular ECG by leveraging the knowledge from pretrained LLM.
- We propose a cardiovascular disease diagnosis and automatic ECG diagnosis report generation approach by transferring the knowledge from LLM to the cardiac ECG domain.
- We introduce an additional learning objective based on Optimal Transport distance, which empowers the model to learn the distribution between ECG and language embedding.
- Our method can generate high-quality cardiac diagnosis reports and achieve competitive zero-shot classification performance even compared with supervised baselines, proving the feasibility of using LLM to enhance research and applications in the cardiac domain.

9.2 Related Work

Cardiovascular Diagnosis via ECG The 12-lead ECG is derived from 10 electrodes placed on the surface of the skin [46]. An ECG works by recording electrical activity corresponding to the heartbeat muscle contractions [41]. Although computerized interpretations of ECGs are widely used, automated approaches have not yet matched the quality of expert cardiologists, leading to poor patient outcomes or even fatality [42].

Deep Learning in ECG Deep learning approaches have been rapidly adopted across a wide range of fields due to their accuracy and flexibility but require large labeled training sets. With the development in machine learning, many models have been applied to ECG disease detection [125, 211, 213, 320, 359, 370, 439, 575]. [13] predicted acute myocardial ischemia in patients with chest pain with a fusion voting method. [6, 297] proposed a nine-layer deep convolutional neural network (CNN) to classify heartbeats in the MIT-BIH Arrhythmia database. [418] estimate a patient’s risk of cardiovascular death after an acute coronary syndrome by a multiple instance learning framework. Recently, [426] proposed models based on SincNet [376] and used entropy-based features for cardiovascular disease classification. The transformer model has also recently been adopted in several ECG applications, i.e., arrhythmia classification, abnormalities detection, stress detection, etc [31, 50, 314, 432, 496, 521].

LLM in Healthcare [562] reviewed existing studies concerning NLP for smart healthcare. [526] developed a large pretrained clinical language model using transformer architecture. [437] showed that using patient representation schemes inspired by techniques in LLM can increase the accuracy of clinical prediction models.

Multimodal Learning in Healthcare Applications Many previous works have explored multimodal learning to boost performance in clinical healthcare applications, i.e., affective computing for depression disease detection and so on [146, 266, 267, 350, 352, 354, 356]. [266, 267, 352, 354, 356] explored the inner correlation between different modalities. [28] investigated the demographics, showing that the subject’s individual characteristics can also be involved in robustness and personalized design. [350] investigated the relationship between computational vision models and computational neuroscience. [146, 167] explored the connectivity between natural language and EEG signals.

9.3 Proposed Method

Problem Formulation We formulate the problem as generating cardiovascular diagnosis reports through pretrained LLMs. Given ECG signals $x = [x_1, x_2, \dots, x_t]$, our goal is to take advantage of the knowledge from LLM and learn a generated text embedding $L = [L_1, L_2, \dots, L_m]$, which can then be decoded into natural language as reports or directly used for disease classification.

Model Architecture The model architecture is shown in Figure 9.2, The ECG inputs are processed by hierarchical transformer encoders [468] to obtain transformed ECG embeddings $X = [X_1, X_2, \dots, X_n]$. Then we adopt a pretrained LLM to transform the ECG embeddings into language embeddings $L = [L_1, L_2, \dots, L_m]$. For the learning objective, we use expert reports to formalize the learning loss, which includes a new loss based on OT in addition to the traditional cross-entropy loss. The learning objective is to update the transformer encoders, which can be interpreted as a sequence-to-sequence mapping from ECG embeddings X to sentence embeddings L . After the learning process, the learned embedding L should be capable of conducting downstream applications. The transform architecture has been introduced in Chapter 2. For the output, we use a 1D convolutional layer and softmax layer to calculate the final output.

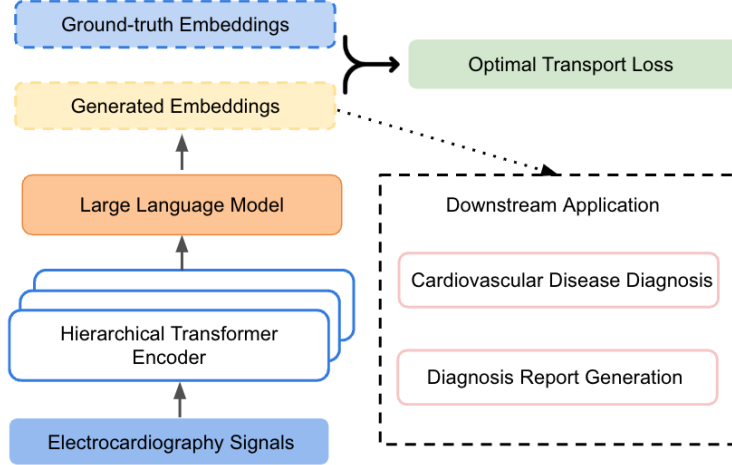


Figure 9.2: The architecture of our model. The Transformer encoder takes input ECG to generate ECG features as the input to LLM, where LLM transforms it into generated embeddings. An OT-based loss objective is formulated on generated embeddings and ground-truth embeddings for the model update.

Downstream Applications For the downstream applications, we first consider a classification problem that uses the embeddings L for cardiovascular disease diagnosis. In addition, we consider a text generation task by decoding the output embeddings L into a cardiovascular report.

Optimal Transport Loss OT is the problem of transporting mass between two discrete distributions supported on latent feature space \mathcal{X} . Let $\mu = \{\mathbf{x}_i, \mu_i\}_{i=1}^n$ and $\nu = \{\mathbf{y}_j, \nu_j\}_{j=1}^m$ be the distributions of generated embeddings and ground-truth embeddings, where $\mathbf{x}_i, \mathbf{y}_j \in \mathcal{X}$ denotes the spatial locations and μ_i, ν_j , respectively, denoting the non-negative masses. Without loss of generality, we assume $\sum_i \mu_i = \sum_j \nu_j = 1$. $\pi \in \mathbb{R}_+^{n \times m}$ is a valid transport plan if its row and column marginals match μ and ν , respectively, which is $\sum_i \pi_{ij} = \nu_j$ and $\sum_j \pi_{ij} = \mu_i$. Intuitively, π transports π_{ij} units of mass at location \mathbf{x}_i to new location \mathbf{y}_j . Such transport plans are not unique, and one often seeks a solution $\pi^* \in \Pi(\mu, \nu)$ that is most preferable in other ways, where $\Pi(\mu, \nu)$ denotes the set of all viable transport plans. OT finds a solution that is most cost-effective w.r.t. cost function $C(\mathbf{x}, \mathbf{y})$:

$$\mathcal{D}(\mu, \nu) = \sum_{ij} \pi_{ij}^* C(\mathbf{x}_i, \mathbf{y}_j) = \inf_{\pi \in \Pi(\mu, \nu)} \sum_{ij} \pi_{ij} C(\mathbf{x}_i, \mathbf{y}_j) \quad (9.1)$$

where $\mathcal{D}(\mu, \nu)$ is known as OT distance. $\mathcal{D}(\mu, \nu)$ minimizes the transport cost from μ to ν w.r.t. $C(\mathbf{x}, \mathbf{y})$. When $C(\mathbf{x}, \mathbf{y})$ defines a distance metric on \mathcal{X} , and $\mathcal{D}(\mu, \nu)$ induces a distance metric on the space of probability distributions supported on \mathcal{X} , it becomes the Wasserstein Distance (WD). We use WD as one loss objective, in addition to the standard cross-entropy loss, for the model update.

9.4 Dataset and Preprocessing

Dataset We conduct the experiments on the PTB-XL dataset [472], which contains clinical 12-lead ECG signals of 10-second length. There are five conditions in total, including Normal

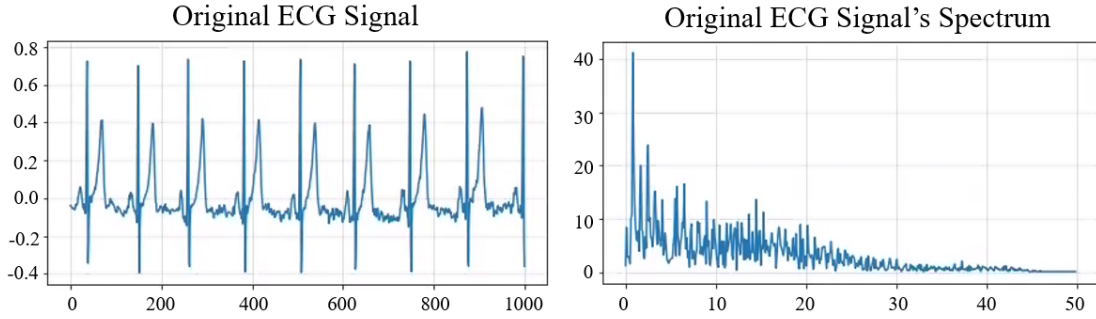


Figure 9.3: ECG data in the format of time series and spectrum.

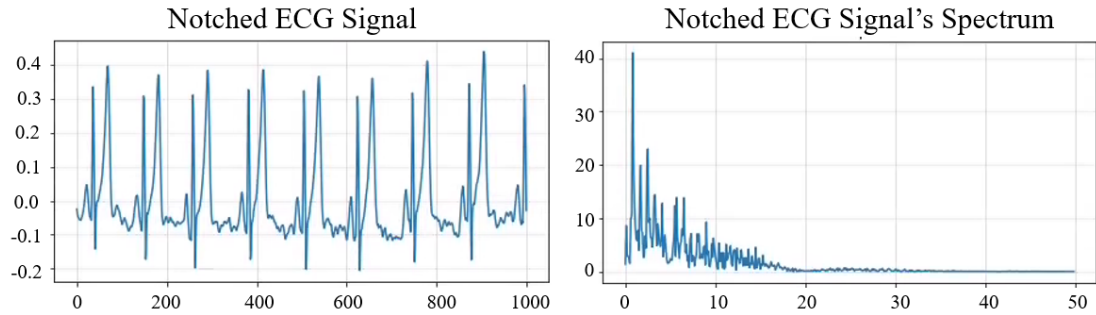


Figure 9.4: Filtered ECG data in the format of time series and spectrum.

ECG (NORM), Myocardial Infarction (MI), ST/T Change (STTC), Conduction Disturbance (CD), and Hypertrophy (HYP). The waveform files are stored in WaveForm DataBase (WFDB) format with 16-bit precision at a resolution of $1\mu\text{V}/\text{LSB}$ and a sampling frequency of 100Hz. The ECG statements conform to the SCP-ECG standard and cover diagnostic, form, and rhythm statements.

Preprocessing The raw ECG signals are first processed by the WFDB library [507] and Fast Fourier transform (FFT) to process the time series data into the spectrum, which is shown in Figure 9.3. Then we perform n-points window filtering to filter the noise within the original ECG signals and adopt notch processing to filter power frequency interference (noise frequency: 50Hz, quality factor: 30). The ECG signals are segmented by dividing the 10-second ECG signals into individual ECG beats. We first detect the R peaks of each signal by ECG detectors [343], and then slice the signal at a fixed-sized interval on both sides of the R peaks to obtain individual beats. Examples of the filtered ECG signal results after n-points window filtering, notch processing, R peak detection, and segmented ECG beats are shown in Figures 9.4,9.5,9.6.

Feature Extraction Instead of directly using the time-series signals, we extract time domain and frequency domain features to better represent ECG signals. The time-domain features include: maximum, minimum, range, mean, median, mode, standard deviation, root mean square, mean square, k-order moment and skewness, kurtosis, kurtosis factor, waveform factor, pulse factor, and margin factor. The frequency-domain features include: FFT mean, FFT variance, FFT entropy, FFT energy, FFT skew, FFT kurt, FFT shape mean, FFT shape std, FFT shape skew, FFT shape kurt. The function of each component is shown in Table 9.1.

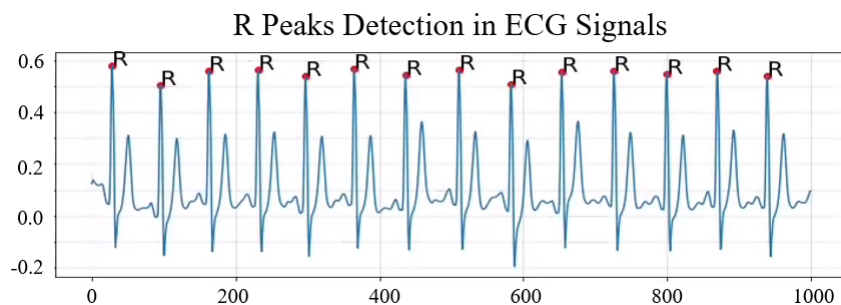


Figure 9.5: Detecting R peaks in the ECG signals.

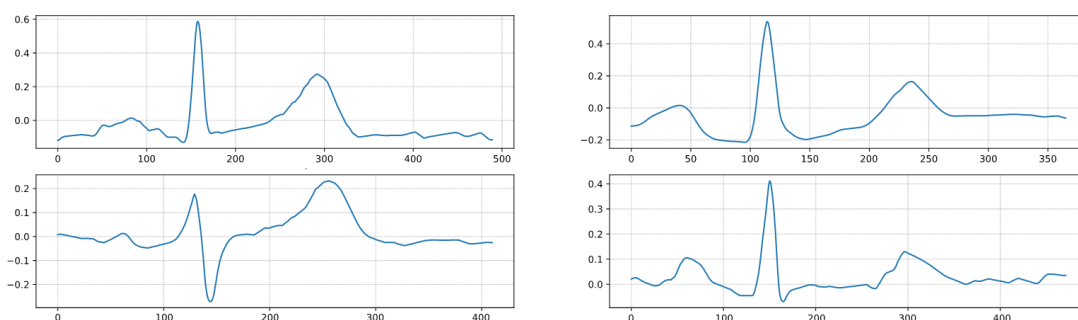


Figure 9.6: Extracted ECG beats divided by R peaks.

Table 9.1: ECG statistical features in the frequency domain.

Feature Symbol	Formula
Z_1	$\frac{1}{N} \sum_{k=1}^N F(k)$
Z_2	$\frac{1}{N-1} \sum_{k=1}^N (F(k) - Z_1)^2$
Z_3	$-1 \times \sum_{k=1}^N \left(\frac{F(k)}{Z_1 N} \log_2 \frac{F(k)}{Z_1 N} \right)$
Z_4	$\frac{1}{N} \sum_{k=1}^N (F(k))^2$
Z_5	$\frac{1}{N} \sum_{k=1}^N \left(\frac{F(k) - Z_1}{\sqrt{Z_2}} \right)^3$
Z_6	$\frac{1}{N} \sum_{k=1}^N \left(\frac{F(k) - Z_1}{\sqrt{Z_2}} \right)^4$
Z_7	$\frac{\sum_{k=1}^N (f(k) - F(k))}{\sum_{k=1}^N F(k)}$
Z_8	$\sqrt{\frac{\sum_{k=1}^N [(f(k) - Z_6)^2 F(k)]}{\sum_{k=1}^N F(k)}}$
Z_9	$\frac{\sum_{k=1}^N [(f(k) - F(k))^3 F(k)]}{\sum_{k=1}^N F(k)}$
Z_{10}	$\frac{\sum_{k=1}^N [(f(k) - F(k))^4 F(k)]}{\sum_{k=1}^N F(k)}$

Table 9.2: Statistics of the processed ECG data.

Category	Patients	Percentage	Beats	Percentage
NORM	9528	34.2%	28419	36.6%
MI	5486	19.7%	10959	14.1%
STTC	5250	18.9%	8906	11.5%
CD	4907	17.6%	20955	27.0%
HYP	2655	9.5%	8342	10.8%

9.5 Experiments

9.5.1 Experimental Settings

Data and Model The dimension of the processed ECG is 864, including 600 ECG signals and 264 time & frequency domain features. Experiments are conducted on two NVIDIA A6000 GPUs.

Tasks To evaluate the learned embeddings from ECG signals, we test the performance on two downstream applications: automatic cardiac report generation as a text generation (TG) task, and zero-shot cardiac disease detection (DD) as a multi-class classification task.

Evaluation For text generation evaluation, we adopt the BLEU [329], ROUGE [257], Meteor [27], and BertScore [554] as evaluation metrics. We report the standard classification evaluation metrics for zero-shot cardiac disease detection: accuracy, AUCROC, and F-1 score.

Table 9.3: Comparisons of different backbones on Text generation (TG) and Disease detection (DD). (BERT as LLM)

Different backbones + BERT as LLM	Text generation (TG)						Disease detection (DD)		
	BLEU-1(%)	ROUGE-1(%)			Meteor(%)	BertScore(%)	Acc	AUCROC	F-1
		P	R	F					
MLP [393]	22.24	17.68	22.63	18.11	14.27	84.68	0.71	0.89	0.57
LSTM [164]	19.74	19.76	18.83	17.99	19.54	84.74	0.73	0.89	0.55
ResNet [156]	21.14	20.35	30.67	25.08	19.55	86.88	0.70	0.86	0.59
Transformer [468]	26.93	25.35	35.67	28.08	21.23	88.90	0.77	0.92	0.68

Table 9.4: Comparisons with supervised baselines (DD).

Supervised learning baselines	Acc	AUROC	F-1
Transformer [575]	0.75	0.843	0.575
CNN [581]	0.72	0.877	0.611
SincNet [376]	0.73	0.84	0.6
Contrastive Learning [227]	–	0.722	–
CNN + Entropy [581]	0.76	0.910	0.68
Ours _{BERT}	0.77	0.92	0.68

Table 9.5: Examples of comparison on generated reports (marked as Predicted-X) and ground-truth reports (marked as GT-X).

Backbone	Reports
GT-1	“sinus rhythm left type peripheral low voltage”
Predicted-1	“ventricular arrhythmia flatfar arrhythmia”
GT-2	“sinus rhythm incomplete right block otherwise normal ekg”
Predicted-2	“ventricularear extrasystole block sinus rhythm or normal.”

Table 9.6: Comparisons of different LLMs on Text generation (TG) and Disease detection (DD). (Transformer as the encoder).

Different LLMs	Text generation (TG)						Disease detection (DD)		
	BLEU-1(%)	ROUGE-1(%)			Meteor(%)	BertScore(%)	Acc	AUCROC	F-1
		P	R	F					
BERT [86]	26.93	25.35	35.67	28.08	21.23	88.90	0.77	0.92	0.68
BART [235]	27.21	26.12	35.71	29.56	24.51	89.61	0.75	0.88	0.68
RoBERTa [269]	27.01	25.31	36.01	27.88	22.41	89.72	0.77	0.89	0.70
BioClinical BERT [12]	27.91	25.41	36.33	28.42	23.54	87.21	0.78	0.89	0.71
PubMed BERT [135]	27.89	25.21	35.97	27.70	24.00	88.56	0.77	0.88	0.69
BioDischargeSummary BERT [12]	26.81	25.32	35.66	28.10	21.19	88.90	0.73	0.85	0.66

9.5.2 Results

In Table 9.3, we show the performance of both text generation and disease detection tasks with different backbone models as baselines. We find that the Transformer encoder outperforms other backbones, i.e., MLP, LSTM, and ResNet, showing Transformer encoder could be a good selection as the feature extractor.

In Table 9.4, we show the performance of our zero-shot disease detection approach, compared with supervised baselines. Even though our method is in the zero-shot setting, we can already achieve the same performance with state-of-the-art supervised learning methods, demonstrating that the transferred ECG representation from LLM is already good for practical usage. We also showed some examples of generated reports compared with ground-truth reports in Table 9.5.

9.5.3 Ablation Study

Different LLM To further analyze the components, we conduct ablation studies on different LLMs and the number of transformer layers (with BERT as LLM). Table 9.6 shows the results of different LLMs for the text generation and disease detection tasks. We found that all LLMs showed good performance in both tasks, demonstrating that knowledge can be transferred from the language domain to the cardiac domain without constraints. BART shows good performance in the text generation task, while BioClinical BERT shows better performance in the disease detection task, though the variation between different LLMs is not large.

Transformer Layers To evaluate the impact of the number of transformer layers, we conduct additional experiments with different transformer layers, and the results are shown in Table 9.11.

Table 9.7: Comparisons with different backbones on the text generation task, where BERT is used as LLM.

Backbone	BLEU-1(%)	ROUGE-1(%)			Meteor(%)	BertScore
		P	R	F		
MLP	18.16	16.19	13.71	14.48	12.11	80.77
LSTM	19.72	19.67	18.83	17.99	19.54	84.73
Resnet	21.15	20.35	20.67	24.08	19.55	85.22
Transformer	24.51	23.22	30.81	26.19	20.02	85.44

Table 9.8: Comparisons with different backbones on the disease detection task, where BERT is used as LLM.

Backbone	Acc	AUCROC	F-1
MLP	0.69	0.77	0.49
LSTM	0.71	0.82	0.59
Resnet	0.70	0.83	0.55
Transformer	0.75	0.81	0.60

Table 9.9: Comparisons of different number of transformer layers on the text generation task, where BERT is used as LLM.

Layers	BLEU-1(%)	ROUGE-1(%)			Meteor(%)	BertScore(%)
		P	R	F		
1	25.52	19.10	27.65	21.43	20.11	86.52
2	24.21	20.00	28.75	23.90	20.32	84.66
3	23.44	20.44	27.21	24.81	19.99	84.63
4	23.17	20.99	28.01	24.44	20.18	87.65
5	25.69	24.75	34.81	27.59	21.03	87.33

Table 9.10: Comparisons of different numbers of transformer layers on the disease detection task, where BERT is used as LLM..

Num of Layers	Acc	AUCROC	F-1
1	0.62	0.79	0.51
2	0.74	0.80	0.60
3	0.71	0.82	0.59
4	0.72	0.83	0.61
5	0.75	0.88	0.64

Table 9.11: Ablation study of different transformer layers.

Layers	Text generation (TG)					Disease detection (DD)			
	BLEU-1(%)	ROUGE-1(%)			Meteor(%)	BertScore(%)	Acc	AUCROC	F-1
		P	R	F					
1	25.81	20.36	30.72	23.12	21.38	83.58	0.69	0.83	0.59
2	24.77	19.22	28.55	24.51	20.44	82.89	0.72	0.81	0.61
3	25.44	20.44	27.21	24.81	19.99	84.63	0.75	0.80	0.62
4	25.12	21.36	30.88	25.76	22.68	86.35	0.74	0.80	0.64
5	26.93	25.35	35.67	28.08	21.23	88.90	0.77	0.92	0.68

We find that more layers could lead to better representations, achieving better performance for downstream applications.

ECG Time Series Signals Only For the results above, we use ECG signals along with ECG time & frequency domain features as inputs. To compare the performance, we also conduct the experiments by only using ECG signals as inputs, with no time & frequency domain features. This set of experiments can be considered an additional ablation study for the inputs. The results are shown in Tables 9.7, 9.8, 9.9, 9.10.

Compare Table 9.7 & 9.8 with Table 9.3, we can find that the performance of only using ECG signals as inputs is lower than combining time & frequency features as inputs in both text generation and disease detection tasks, which demonstrates that incorporating time & frequency features is useful for capturing the characteristics of ECG and can lead to better representations through LLM.

In Tables 9.9, 9.10, the transformer backbone performs the best compared to others in both disease detection and text generation tasks, showing that more layers could lead to better representations, achieving better performance for downstream applications. In addition, compared with Table 9.11, we can find that the performance in Tables 9.9 and 9.10 are lower than the ones in Table 9.11, which also proved the same findings that adding time & frequency features is useful for learning the cardiac ECGs.

9.5.4 Limitations

Due to the constrain of the available datasets, we only conduct experiments on the PTB-XL dataset, which is the current largest ECG dataset that contains high-quality clinical ECG signals and cardiac reports by experienced cardiologists.

We understand that collecting high-quality clinical data is much more complicated and time-consuming than collecting other data from online resources, like images since it requires expert domain knowledge and is limited by many privacy regulations. We are working with cardiologists, hospitals, and clinical research labs, hope we can release a new dataset to provide additional materials for this research direction.

9.6 Conclusion

In this chapter, we bridge the gap between LLMs and cardiovascular ECG by transferring knowledge of LLMs into the cardiovascular domain. The transferred knowledge embeddings can be used for downstream applications, including cardiovascular disease diagnosis and automatic ECG diagnosis report generation. Our results demonstrate the effectiveness of knowledge transfer, as the proposed method shows excellent performance in both downstream tasks, whereas our zero-shot classification approach even achieved competitive performance with supervised learning baselines, showing the feasibility of using LLM to enhance applications in the cardiovascular domain.

Chapter 10

Inner Alignment between Brain Signals and Human Languages

In addition to the ECG signal in Chapter 9, brain signals, such as Electroencephalography (EEG), and human languages have been widely explored independently for many downstream tasks, however, the connection between them has not been well explored. In this study, we explore the relationship and dependency between EEG and language. To study at the representation level, we introduced **MTAM**, a **Multimodal Transformer Alignment Model**, to observe coordinated representations between the two modalities. We use various relationship alignment-seeking techniques, such as Canonical Correlation Analysis and Wasserstein Distance, as loss functions to transfigure features. On downstream applications, sentiment analysis and relation detection, we achieve new state-of-the-art results on two datasets, ZuCo and K-EmoCon. Our method achieve an F1-score improvement of 1.7% on K-EmoCon and 9.3% on Zuco datasets for sentiment analysis, and 7.4% on ZuCo for relation detection. In addition, we provide interpretations of the performance improvement: (1) feature distribution shows the effectiveness of the alignment module for discovering and encoding the relationship between EEG and language; (2) alignment weights show the influence of different language semantics as well as EEG frequency features; (3) brain topographical maps provide an intuitive demonstration of the connectivity in the brain regions.

10.1 Introduction

Brain activity is an important parameter in furthering our knowledge of how human language is represented and interpreted [83, 304, 379, 409, 459, 492, 499]. Researchers from domains such as linguistics, psychology, cognitive science, and computer science have made large efforts in using brain-recording technologies to analyze cognitive activity during language related tasks and observed that these technologies have added value in terms of understanding language [438].

Basic linguistic rules seem to be effortlessly understood by humans in contrast to machinery. Recent advances in NLP models [468] have enabled computers to maintain long and contextual information through self-attention mechanisms. This attention mechanism has been maneuvered to create robust language models but at the cost of tremendous amounts of data [43, 86, 235, 269, 527]. Although performance has significantly improved by using modern NLP models, they are still seen to be suboptimal compared to the human brain. [38] argues that the language-only approach in training

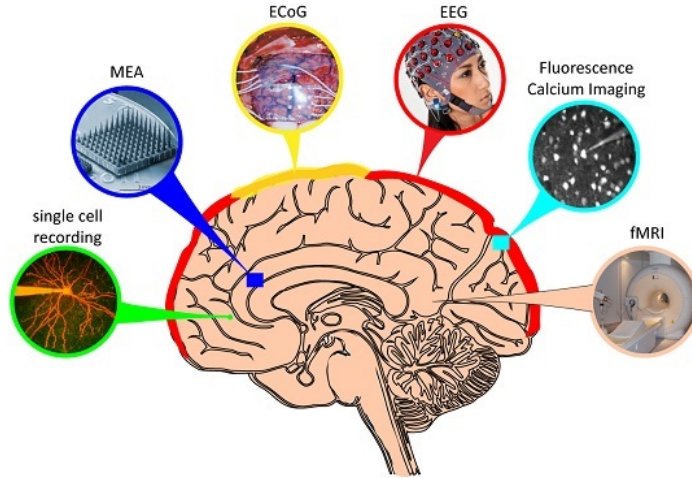


Figure 10.1: Commonly used techniques for recording brain activity [1].

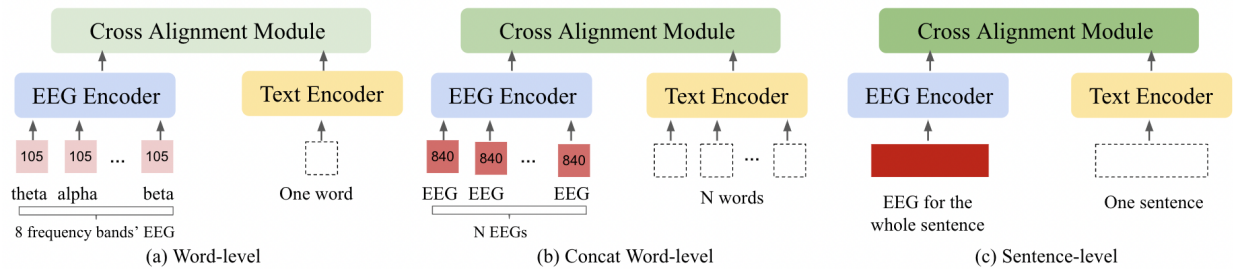


Figure 10.2: Three paradigms of EEG and language alignment.

reaches a point of diminishing returns and extra-linguistic factors are needed in comprehending language through computational procedures.

To combat the limitations of unimodal approaches in NLP, [259] encouraged scholars to gather multimodal data to accelerate comprehension and generalization of natural language in machinery. A popular multimodal framework encodes features from different modalities into a common latent space and maps the latent representations to a specified task [185]. [185] proves that learning with multiple modalities attains a smaller population risk and an accurate estimate of latent space representations. Most existing work in multimodal learning combines a variation of language, vision, and speech signals to perform a wide range of tasks, including but not limited to automatic image and video tagging, speech recognition, and identity classification [82, 98, 214, 529]. More recently, physiological signals have gained attention in the NLP multimodal realm due to their abundance of information and proven practicality across many assignments [166]. In the context of modeling human-like learning phenomena for language, it is instinctively appealing to leverage physiological signals. However, in practice, wielding multiple modalities, including physiological signals and language, is often challenging due to the heterogeneity and contingencies found in the data [300]. [488] proposed a method for EEG-To-Text decoding and achieved great performance, however, the real relationship and connectivity between EEG and language are not well studied.

In this study, we explore the relationship and dependencies of EEG and language. We apply EEG, a popularized routine in cognitive research, for its accessibility and practicality, along with language to discover connectivity. Our contributions are summarized as follows:

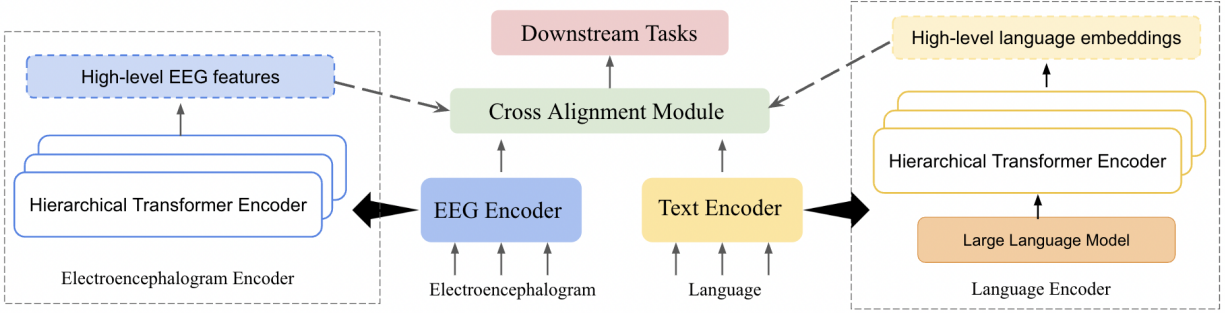


Figure 10.3: The architecture of our model, where EEG and language features are coordinately explored by two encoders. The EEG encoder and language encoder are shown on the left and right, respectively. The cross-alignment module is used to explore the connectivity and relationship within two domains, while the transformed features are used for downstream tasks.

- To the best of our knowledge, we are the first to explore the fundamental relationship and connectivity between EEG and language through computational multimodal alignment methods.
- We introduce **MTAM**, a **Multimodal Transformer Alignment Model**, that learns coordinated representations by hierarchical transformer encoders. The transformed representations showed tremendous performance improvements and state-of-the-art results in downstream applications, i.e., sentiment analysis and relation detection, on two datasets, ZuCo and K-EmoCon.
- We carry out experiments with multiple alignment mechanisms, i.e., canonical correlation analysis and Wasserstein distance, and demonstrated that relation-seeking loss functions are helpful in downstream tasks.
- We provide interpretations of the performance improvement by visualizing the original feature distribution and the transformed feature distribution, showing the effectiveness of the alignment module for discovering and encoding the relationship between EEG and language.
- Our findings on word-level and sentence-level EEG-language alignment show the influence of different language semantics as well as EEG frequency features, which provided additional explanations.
- The brain topographical maps delivered an intuitive demonstration of the connectivity of EEG and language response in the brain regions, which issues a physiological basis for our discovery.

10.2 Related Work

Multimodal Learning of EEG and Other Domains [34] used EMG signals jointly with EEG in a bi-autoencoder architecture and increased accuracies for sentiment analysis. [29] integrated ECG and EEG signals in a human identification task, where fused classifiers produced the highest score. [28, 266, 267] extracted correlated features between EEG and eye movement data for emotion classification, showing transformed features are more homogeneous and discriminative. [323] fed fNIRS and EEG to decode bimanual grip force and resulted in increased performance compared to

single modality models. There are also efforts to find correlations between EEG and visual stimulus frequencies [398]. A common theme occurring among these works showed EEG paired with other domains can boost performance.

Multimodal Learning of Language and Other Brain Signals Recently, language and cognitive data were also used together in multimodal settings to complete desirable tasks [165, 166, 167, 488]. [493] used a recurrent neural network to perform word alignment between MEG activity and the generated word embeddings. [460] utilized word-level MEG and fMRI recordings to compare word embeddings from large language models. [409] used MEG and fMRI data to fine-tune a BERT language model [86] and found that the relationships between these two modalities were generalized across participants. [180] leveraged CT images and text from electronic health records to classify pulmonary embolism cases and observed that the multimodal model with late fusion achieved the best performance. [303] found semantic categories between MEG and language. However, the relationship between language and EEG has not been explored before.

Multimodal Learning of EEG and Language [144] related EEG signals to the states of a neural phrase structure parser and showed that through EEG signals, models were correlating syntactic properties to a specific genre of text. [109] applied EEG signals to predict specific values of each dimension in a word vector through regression models. [488] used word-level EEG features to decode corresponding text tokens through a sequence-to-sequence framework. [167] focused on a multimodal approach by utilizing a combination of EEG, eye-tracking, and text data to improve NLP tasks. They used a variation of LSTM and CNN to decode the EEG features but did not explore the relationship between EEG and language. Their proposed multimodal framework follows the bi-encoder approach [67] where the two modalities are encoded separately [167].

10.3 Proposed Method

10.3.1 Overview of Model Architecture

The architecture of our model is shown in Figure 10.3. The bi-encoder architecture is helpful in projecting embeddings into vector space for methodical analysis [67, 167, 266]. Thus in our study, we adopt the bi-encoder approach to effectively reveal hidden relations between language and EEG. The **MTAM**, Multimodal Transformer Alignment Model, contains several modules. We use a dual-encoder architecture, where each view contains hierarchical transformer encoders. The inputs of each encoder are EEG and language, respectively. For EEG hierarchical encoders, each encoder shares the same architecture as the encoder module in [468]. In the current literature, researchers assume that the brain acts as an encoder for high-dimensional semantic representations [72, 122, 488]. Based on this assumption, the EEG signals act as low-level embeddings. By feeding it into its respective hierarchical encoder, we extract transformed EEG embeddings as input for the cross-alignment module. As for the language path, the language encoder is slightly different from the EEG encoder. We first process the text with a pretrained large language model (LLM) to extract text embeddings and then use hierarchical transformer encoders to transform the raw text embeddings into high-level features. The mechanism of the cross-alignment module is to explore the inner relationship between EEG and language through a connectivity-based loss function. In

our study, we investigate several alignment methods, i.e., **CCA** and **WD**. The output features from the cross-alignment module can be used for downstream applications. The details of each part will be introduced in the following sections.

10.3.2 Hierarchical Transformer Encoders

Let $X_e \in \mathbb{R}^{D_e}$ and $X_t \in \mathbb{R}^{D_t}$ be the two normalized input feature matrices for EEG and text, respectively, where D_e and D_t describes the dimensions of the feature matrices. To encode the two feature vectors, we feed them to their hierarchical transformer encoders: $V_e = E_e(X_e; W_e)$; $V_t = E_t(X_t; W_t)$, where E_e and E_t denotes the separate encoders, V_e and V_t symbolizes the outputs for the transformed low-level features and W_e and W_t denotes the trainable weights for EEG and text respectively. The outputs of these two encoders can be further expanded by stating $V_e = [v_e^1, v_e^2, v_e^3, \dots, v_e^n] \in \mathbb{R}^n$ and $V_t = [v_t^1, v_t^2, v_t^3, \dots, v_t^k] \in \mathbb{R}^k$, where n and k denotes the number of instances in a given output vector and v_e^n and v_t^k denotes the instance itself. The transformer encoder we use in this chapter is the same as the model architecture design in Chapter 9, and the transformer basic has been introduced in Chapter 2 as well.

10.3.3 Cross Alignment Module

As shown in Figure 10.2, there are three paradigms of EEG and language alignment. For word level, the EEG features are divided by each word, and the objective of the alignment is to find the connectivity of different frequencies with the corresponding word. For the concat-word level, the 8 frequencies' EEG features are concatenated as a whole and then concatenated again to match the corresponding sentence, so the alignment is to find out the relationship within the sentence. As for sentence level, the EEG features are calculated as an average over the word-level EEG features. There is no boundary for the word, so the alignment module tries to encode the embeddings as a whole and explore the general representations. In the Cross Alignment Module (CAM), we introduced a new loss function in addition to the original cross-entropy loss. The new loss is based on **CCA** [16] and Optimal Transport (Wasserstein Distance). As in [16], CCA aims to concurrently learn the parameters of two networks to maximize the correlation between them. **WD**, which originates from **OT**, has the ability to align embeddings from different domains to explore the relationship [54].

Canonical Correlation Analysis **CCA** is a method for exploring the relationships between two multivariate sets of variables. It learns the linear transformation of two vectors to maximize the correlation between them, which is used in many multimodal problems [16, 129, 352]. In this chapter, we apply CCA to capture the cross-domain relationship. Let low-level transformed EEG features be V_e and low-level language features be L_t . We assume $(V_e, V_t) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2}$ has covariances $(\Sigma_{11}, \Sigma_{22})$ and cross-covariance Σ_{12} . CCA finds pairs of linear projections of the two views, $(w_1' V_e, w_2' V_t)$ that are maximally correlated:

$$(w_1^*, w_2^*) = \operatorname{argmax}_{w_1, w_2} \operatorname{corr}(w_1' V_e, w_2' V_t) = \operatorname{argmax}_{w_1, w_2} \frac{w_1' \Sigma_{12} w_2}{\sqrt{w_1' \Sigma_{11} w_1 w_2' \Sigma_{22} w_2}} \quad (10.1)$$

In our study, we modified the structure of [16] while honoring its duty by replacing the neural networks with Transformer encoders. w_1^* and w_2^* denote the high-level, transformed weights from

the low-level text and EEG features, respectively.

Wasserstein Distance WD is introduced in OT, which is a natural type of divergence for registration problems as it accounts for the underlying geometry of the space, and has been used for multimodal data matching and alignment tasks [54, 81, 230, 360, 540, 575]. In Euclidean settings, OT introduces WD $\mathcal{W}(\mu, \nu)$, which measures the minimum effort required to “displace” points across measures μ and ν , where μ and ν are values observed in the empirical distribution. In our setting, we compute the temporal-pairwise Wasserstein Distance on EEG features and language features, which are $(\mu, \nu) = (V_e, V_t)$. For simplicity without loss of generality, assume $\mu \in P(\mathbb{X})$ and $\nu \in P(\mathbb{Y})$ denote the two discrete distributions, formulated as $\mu = \sum_{i=1}^n u_i \delta_{x_i}$ and $\nu = \sum_{j=1}^m v_j \delta_{y_j}$, with δ_x as the Dirac function centered on x . $\Pi(\mu, \nu)$ denotes all the joint distributions $\gamma(x, y)$, with marginals $\mu(x)$ and $\nu(y)$. The weight vectors $u = \{u_i\}_{i=1}^n \in \Delta_n$ and $v = \{v_i\}_{i=1}^m \in \Delta_m$ belong to the n - and m -dimensional simplex, respectively. The WD between the two discrete distributions μ and ν is defined as:

$$\mathcal{WD}(\mu, \nu) = \inf_{\gamma \in \Pi(\mu, \nu)} \mathbb{E}_{(x, y) \sim \gamma} [c(x, y)] = \min_{\mathbf{T} \in \Pi(\mathbf{u}, \mathbf{v})} \sum_{i=1}^n \sum_{j=1}^m T_{ij} \cdot c(x_i, y_j) \quad (10.2)$$

where $\Pi(\mathbf{u}, \mathbf{v}) = \{\mathbf{T} \in \mathbb{R}_+^{n \times m} | \mathbf{T} \mathbf{1}_m = \mathbf{u}, \mathbf{T}^\top \mathbf{1}_n = \mathbf{v}\}$, $\mathbf{1}_n$ denotes an n -dimensional all-one vector, and $c(x_i, y_j)$ is the cost function evaluating the distance between x_i and y_j .

Loss Objective The loss objective for the CAM module can be formalized as: $Loss = l_{CE} + \alpha_1 l_{CCA} + \alpha_2 l_{WD}$, where $\alpha_i \in \{0, 1\}$, $i \in (1, 2)$ controls the weights of different parts of alignment-based loss objective. The loss objective for the CAM module can be formalized as: $Loss = l_{CE} + \alpha_1 l_{CCA} + \alpha_2 l_{WD}$, where $\alpha_i \in \{0, 1\}$, $i \in (1, 2)$ controls the weights of different parts of alignment-based loss objective.

10.4 Experiments

10.4.1 Downstream Tasks

In this study, we evaluate our method on two downstream tasks: Sentiment Analysis (SA) and Relation Detection (RD) of two datasets: K-EmoCon [331] and ZuCo 1.0/2.0 Dataset [168, 169].

Sentiment Analysis (SA) Given a succession of word-level or sentence-level EEG features and their corresponding language, the task is to predict the sentiment label. The ZuCo 1.0 dataset consists of sentences from the Stanford Sentiment Treebank, which contains movie reviews and their corresponding sentiment label (i.e., positive, neutral, negative) [427]. The K-EmoCon dataset categorizes emotion annotations as valence, arousal, happy, sad, nervous, and angry. For each emotion, the participant labeled the extent of the given emotion felt by following a Likert-scale paradigm. Arousal and valence are rated 1 to 5 (1: very low; 5: very high). Happy, sad, nervous, and angry emotions are rated 1 to 4, where 1 means very low and 4 means very high. The ratings are dominantly labeled as very low and neutral. Therefore to combat class imbalance, we collapse the labels to binary and ternary settings.

Relation Detection (RD) The goal of relation detection (also known as relation extraction or entity association) is to extract semantic relations between entities in a given text. For example, in the sentence, "June Huh won the 2022 Fields Medal.", the relation *AWARD* connects the two entities "June Huh" and "Fields Medal" together. The ZuCo 1.0/2.0 datasets provide the ground truth labels and texts for this task. We use texts from the Wikipedia relation extraction dataset [75] that has 10 relation categories: award, control, education, employer, founder, job title, nationality, political affiliation, visited, and wife [168, 169].

10.4.2 Datasets

K-EmoCon Dataset K-EmoCon [331] is a multimodal dataset including videos, speech audio, accelerometer, and physiological signals during a naturalistic conversation. After the conversation, each participant watched a recording of themselves and annotated their own and partner’s emotions. Five external annotators were recruited to annotate both parties’ emotions, six emotions in total (Arousal, Valence, Happy, Sad, Angry, and Nervous). The NeuroSky MindWave headset captured EEG signals from the left prefrontal lobe (FP1) at a sampling rate of 125 Hz in 8 frequency bands: delta (0.5–2.75Hz), theta (3.5–6.75Hz), low-alpha (7.5–9.25Hz), high-alpha (10–11.75Hz), low-beta (13–16.75Hz), high-beta (18–29.75Hz), low-gamma (31–39.75Hz), and middle-gamma (41–49.75Hz). We used Google Cloud’s Speech-to-Text API to transcribe the audio data into text.

ZuCo Dataset The ZuCo Dataset [168, 169] is a corpus of EEG signals and eye-tracking data during natural reading. The tasks during natural reading can be separated into three categories: sentiment analysis, natural reading, and task-specified reading. During sentiment analysis, the participant was presented with 400 positive, neutral, and negative labeled sentences from the Stanford Sentiment Treebank [427]. The EEG data used in this study can be categorized into sentence-level and word-level features. The sentence-level features are the averaged word-level EEG features for the entire sentence duration. The word-level EEG features are for the first fixation duration (FFD) of a specific word, meaning when the participant’s eye met the word, the EEG signals were recorded. For both word and sentence-level features, 8 frequency bands were recorded at a sampling frequency of 500 Hz and denoted as the following: theta1 (4-6Hz), theta2 (6.5–8Hz), alpha1 (8.5–10Hz), alpha2 (10.5–13Hz), beta1 (13.5–18Hz), beta2 (18.5–30Hz), gamma1 (30.5–40Hz), and gamma2 (40–49.5Hz).

10.4.3 Experimental Settings

Model Settings The hierarchical transformer encoders follow the standard skeleton from [468], excluding its complexity. To avoid overfitting, we adopt the oversampling strategy for data augmentation [187], which ensures a balanced distribution of classes included in each batch. The train/test/validation splitting is (80%, 10%, 10%) as in [167]. The EEG features are extracted from the datasets in 8 frequency bands and normalized with Z-score according to previous work [95, 103, 397] over each frequency band. To preserve reliability, the word and sentence embeddings are also normalized with Z-scores. We use pre-trained language models to generate text features [86], where all texts are tokenized and embedded using the BERT-uncased-base model. Each sentence has an average length of 20 tokens, so we instantiate a max length of 32 with padding. In the case of word-level, we use an average length of 4 tokens for each word and establish a max

Table 10.1: Comparison with baselines on K-EmoCon dataset for Sentiment Analysis.

Model	Prec	Rec	F1	Acc
MLP-EEG	0.295	0.317	0.222	0.231
MLP-Text	0.263	0.272	0.182	0.180
Bi-LSTM-EEG	0.340	0.354	0.226	0.220
Bi-LSTM-Text	0.241	0.329	0.125	0.224
Transformer-EEG	0.399	0.411	0.405	0.484
Transformer-Text	0.454	0.492	0.472	0.443
ResNet-EEG	0.456	0.389	0.202	0.229
ResNet-Text	0.133	0.348	0.169	0.224
Ours-EEG	0.591	0.516	0.551	0.591
Ours-Text	0.524	0.561	0.509	0.542
Ours-Multimodal	0.739	0.720	0.729	0.733

length of 10 with padding. The token vectors from the four last hidden layers of the pre-trained model are withdrawn and averaged to get a final sentence or word embedding. These embeddings are used during the sentence-level and word-level settings. For the concat word level, we simply concatenate the word embeddings for their respective sentence.

Baselines The area of multimodal learning of EEG and language is not well explored, and to the best of our knowledge, only [167]’s approach was directly comparable to our study. However, to make a fair evaluation, we implemented the following state-of-the-art representative approaches as baselines for verification: MLP [394], Bi-LSTM [134, 570], Transformer [468], and ResNet [156].

10.4.4 Experimental Results and Discussions

In Table 10.1, we show the comparison results of different methods on the K-EmoCon dataset. From Table 10.1, we can see that our method outperforms the other baselines, and the multimodal approach outperforms the unimodal approach, which also demonstrates the effectiveness of our method. In Table 10.2, we show the comparison results of the ZuCo dataset for Sentiment Analysis and Relation Detection, respectively. Our method outperforms all baselines, and the multimodal approach outperforms unimodal approaches, which further demonstrates the importance of exploring the inner alignment between EEG and language.

10.4.5 Ablation Study

To further investigate the performance of different mechanisms in the CAM module, we carry out ablation experiments on the Zuco dataset, and the results are shown in Table 10.3. The combination of CCA and WD performs better compared to using only one mechanism for sentiment analysis and relation detection in all model settings.

We also conduct experiments on word-level, sentence-level, and concat word-level inputs, and the results are also shown in Table 10.3. We observe that word-level EEG features paired with their respective word generally outperform sentence-level and concat word-level in both tasks.

K-EmoCon Previous Work To the best of our knowledge, there is no existing work where EEG or text is used for the K-EmoCon dataset. However, other modalities such as audio, video, blood

Table 10.2: Comparison with baselines on Zuco dataset for Sentiment Analysis (SA) and Relation Detection (SD).

Task	Model	Sentence Level				Word Level				Concat Word Level			
		Prec	Rec	F1	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc
Sentiment Analysis	MLP-EEG	0.644	0.637	0.640	0.666	0.602	0.644	0.622	0.630	0.597	0.618	0.607	0.600
	MLP-Text	0.359	0.357	0.357	0.373	0.380	0.388	0.384	0.387	0.210	0.243	0.225	0.228
	Bi-LSTM-EEG	0.675	0.656	0.664	0.666	0.677	0.659	0.668	0.671	0.612	0.609	0.610	0.608
	Bi-LSTM-Text	0.420	0.347	0.380	0.371	0.335	0.326	0.330	0.329	0.341	0.322	0.331	0.329
	Transformer-EEG	0.887	0.879	0.883	0.883	0.832	0.840	0.836	0.881	0.832	0.840	0.836	0.817
	Transformer-Text	0.548	0.546	0.547	0.507	0.527	0.533	0.530	0.582	0.558	0.547	0.552	0.550
	ResNet-EEG	0.687	0.678	0.682	0.683	0.707	0.718	0.712	0.709	0.691	0.689	0.690	0.688
	ResNet-Text	0.214	0.183	0.165	0.222	0.198	0.199	0.198	0.200	0.202	0.211	0.206	0.210
	RNN-Multimodal [167]	—	—	—	—	0.728	0.717	0.714	—	—	—	—	—
	CNN-Multimodal [167]	—	—	—	—	0.738	0.724	0.723	—	—	—	—	—
	Ours-EEG	0.984	0.991	0.989	0.984	0.991	0.997	0.994	0.990	0.891	0.888	0.889	0.890
	Ours-Text	0.850	0.849	0.849	0.817	0.832	0.834	0.833	0.839	0.823	0.881	0.850	0.891
Ours-Multimodal	0.989	0.997	0.993	0.993	0.986	0.977	0.981	0.994	0.966	0.975	0.970	0.978	
Relation Detection	MLP-EEG	0.450	0.455	0.452	0.450	0.463	0.471	0.467	0.463	0.435	0.438	0.436	0.430
	MLP-Text	0.191	0.214	0.192	0.254	0.249	0.286	0.266	0.258	0.228	0.231	0.229	0.230
	Bi-LSTM-EEG	0.552	0.570	0.556	0.549	0.584	0.591	0.587	0.585	0.564	0.541	0.552	0.551
	Bi-LSTM-Text	0.153	0.173	0.149	0.186	0.200	0.199	0.199	0.201	0.182	0.133	0.154	0.148
	Transformer-EEG	0.589	0.517	0.551	0.564	0.399	0.401	0.400	0.421	0.364	0.372	0.368	0.370
	Transformer-Text	0.428	0.487	0.444	0.420	0.488	0.491	0.489	0.487	0.301	0.299	0.300	0.300
	ResNet-EEG	0.514	0.571	0.590	0.558	0.499	0.501	0.500	0.500	0.503	0.519	0.511	0.504
	ResNet-Text	0.314	0.283	0.265	0.322	0.311	0.336	0.323	0.326	0.278	0.289	0.283	0.288
	CNN-Multimodal [167]	—	—	—	—	0.647	0.664	0.650	—	—	—	—	—
	RNN-Multimodal [167]	—	—	—	—	0.652	0.690	0.668	—	—	—	—	—
	Ours-EEG	0.942	0.976	0.959	0.932	0.922	0.938	0.930	0.931	0.900	0.943	0.921	0.922
	Ours-Text	0.743	0.751	0.747	0.732	0.744	0.784	0.763	0.759	0.634	0.686	0.659	0.660
Ours-Multimodal	0.979	0.987	0.979	0.982	0.977	0.980	0.978	0.984	0.969	0.965	0.967	0.969	

volume pulse (BVP), electrodermal activity (EDA), body temperature (TEMP), skin temperature (SKT), accelerometer (ACC) and heart rate (HR) have been used to perform sentiment analysis. As shown in Table 10.4, our model outperforms the previous method, with even less domains data, showing the connectivity between EEG and language and also the advantages of exploring them for downstream applications.

10.4.6 Analysis

In order to interpret the performance improvement, we visualize the original feature distribution and the transformed feature distribution. As shown in Figure 10.4, the transformed feature distribution makes better clusters than the original one. The alignment module reduces the randomness and sparsity, showing the effectiveness of discovering and encoding the relationship between EEG and language.

Furthermore, to understand the alignment between language and EEG, we visualize the alignment weights of word-level EEG-language alignment on the ZuCo dataset. Figure 10.5 and Figure 10.6 show examples of negative & positive sentence word-level alignment, respectively. Figure 10.7 shows the negative and positive sentence-level alignment weights of the ZuCo dataset. In Figure 10.7, we can find that alpha1, beta1, and gamma1 frequency bands show larger different responses between negative and positive sentences.

From the word level alignment in Figure 10.5 and Figure 10.6, beta2 and gamma1 waves are most active. This is consistent with the literature, which shows that gamma waves are seen to be active in detecting emotions [246], and beta waves have been involved in higher-order linguistic

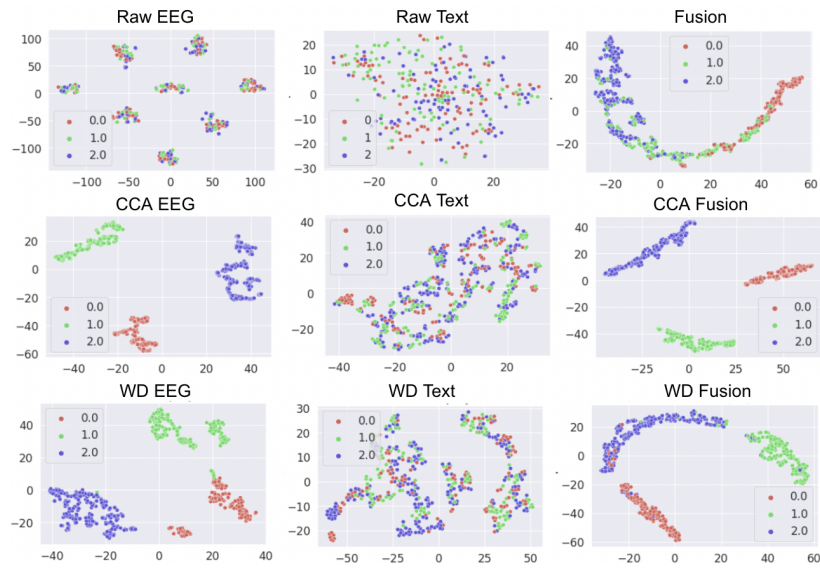


Figure 10.4: TSNE projection comparison of untransformed & transformed features of ZuCo dataset, where different colors represent different classes.

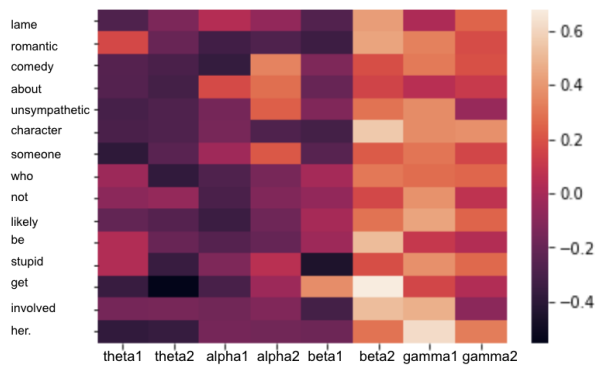


Figure 10.5: Negative word-level alignment.

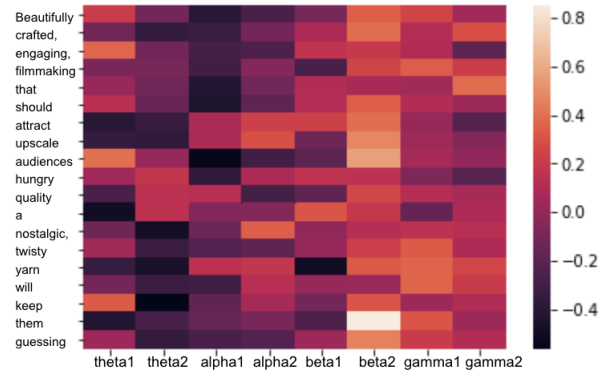


Figure 10.6: Positive word-level alignment.

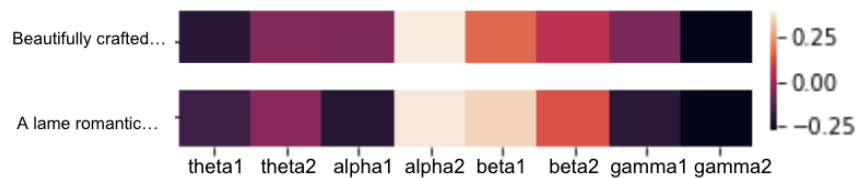


Figure 10.7: Negative and Positive sentence-level alignment of ZuCo dataset.

Table 10.3: Ablation results on the components in the CAM module (best results in bold).

Dataset	Model	Sentence Level				Word Level				Concat Word Level			
		Prec	Rec	F1	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc
ZuCo (SA)	Ours-CCA-Text	0.748	0.746	0.747	0.707	0.701	0.733	0.717	0.769	0.752	0.787	0.769	0.744
	Ours-CCA-EEG	0.984	0.991	0.989	0.984	0.988	0.991	0.989	0.985	0.970	0.975	0.972	0.971
	Ours-CCA-All	0.987	0.956	0.971	0.991	0.989	0.979	0.984	0.991	0.959	0.973	0.966	0.972
	Ours-WD-Text	0.618	0.604	0.611	0.624	0.753	0.747	0.750	0.770	0.740	0.731	0.735	0.733
	Ours-WD-EEG	0.965	0.930	0.942	0.948	0.982	0.999	0.990	0.991	0.888	0.882	0.885	0.867
	Ours-WD-All	0.910	0.862	0.885	0.981	0.985	0.999	0.992	0.994	0.918	0.922	0.985	0.917
	Ours-CCA+WD-Text	0.850	0.849	0.849	0.817	0.832	0.834	0.833	0.839	0.823	0.881	0.850	0.891
	Ours-CCA+WD-EEG	0.979	0.982	0.980	0.983	0.991	0.997	0.994	0.990	0.891	0.888	0.889	0.890
	Ours-CCA+WD-All	0.989	0.997	0.993	0.993	0.986	0.977	0.981	0.994	0.966	0.975	0.970	0.978
ZuCo (RD)	Ours-CCA-Text	0.750	0.749	0.749	0.717	0.698	0.661	0.679	0.650	0.551	0.655	0.599	0.602
	Ours-CCA-EEG	0.851	0.926	0.874	0.863	0.780	0.781	0.780	0.782	0.744	0.801	0.771	0.750
	Ours-CCA-All	0.892	0.930	0.885	0.881	0.911	0.923	0.917	0.904	0.851	0.866	0.858	0.855
	Ours-WD-Text	0.674	0.642	0.658	0.668	0.671	0.624	0.647	0.651	0.597	0.577	0.587	0.599
	Ours-WD-EEG	0.870	0.867	0.868	0.847	0.899	0.908	0.903	0.900	0.891	0.925	0.908	0.900
	Ours-WD-All	0.802	0.857	0.829	0.880	0.898	0.943	0.920	0.914	0.898	0.866	0.882	0.916
	Ours-CCA+WD-Text	0.743	0.751	0.747	0.732	0.744	0.784	0.763	0.759	0.634	0.686	0.659	0.660
	Ours-CCA+WD-EEG	0.942	0.976	0.959	0.932	0.922	0.938	0.930	0.931	0.900	0.943	0.921	0.922
	Ours-CCA+WD-All	0.979	0.987	0.979	0.982	0.977	0.980	0.978	0.984	0.969	0.965	0.967	0.969

Table 10.4: Comparison of performance on K-EmoCon dataset with different physiological signals as inputs on the Sentiment Analysis task.

Model	Modalities	Rec	F1	Acc
CNN + Transformer [364]	Video and Audio	0.693	0.712	0.725
CNN Fusion [87]	ACC, BVP, EDA, TEMP	NA	0.562	0.591
Convolution-augmented Transformer [524]	BCP, EDA, HR, SKT	0.655	0.564	NA
Transformer [524]	BCP, EDA, HR, SKT	0.628	0.518	NA
BiLSTM [524]	BCP, EDA, HR, SKT	0.563	0.473	NA
Ours	Text, EEG	0.720	0.729	0.733

functions (e.g., discrimination of word categories) [167]. [167] found that beta and theta waves were most useful in terms of model performance in sentiment analysis.

We perform an analysis of which EEG feature refined the model’s performance since different neurocognitive factors during language processing are associated with brain oscillations at miscellaneous frequencies. The beta and theta bands have positively contributed the most, which is due to the theta band power expected to rise with increased language processing activity and the band’s relation to semantic memory retrieval [167, 221]. The beta’s contribution can be best explained by the effect of emotional connotations of the text [30, 167].

In Figure 10.8, we visualize the brain topologies with word-level EEG features for important and unimportant words from positive and negative sentences in the ZuCo dataset. We deemed a word important if the definition had a positive or negative connotation. ‘Upscale’ and ‘lame’ are important positive and negative words, respectively, and ‘will’ and ‘someone’ are unimportant positive and negative words, respectively. There are two areas in the brain that are heavily associated with language processing: Broca’s area and Wernicke’s area. Broca’s area is assumed to be located in the left frontal lobe, and this region is concerned with the production of speech [313]. The left posterior superior temporal gyrus is typically assumed as Wernicke’s area, and this locale is involved

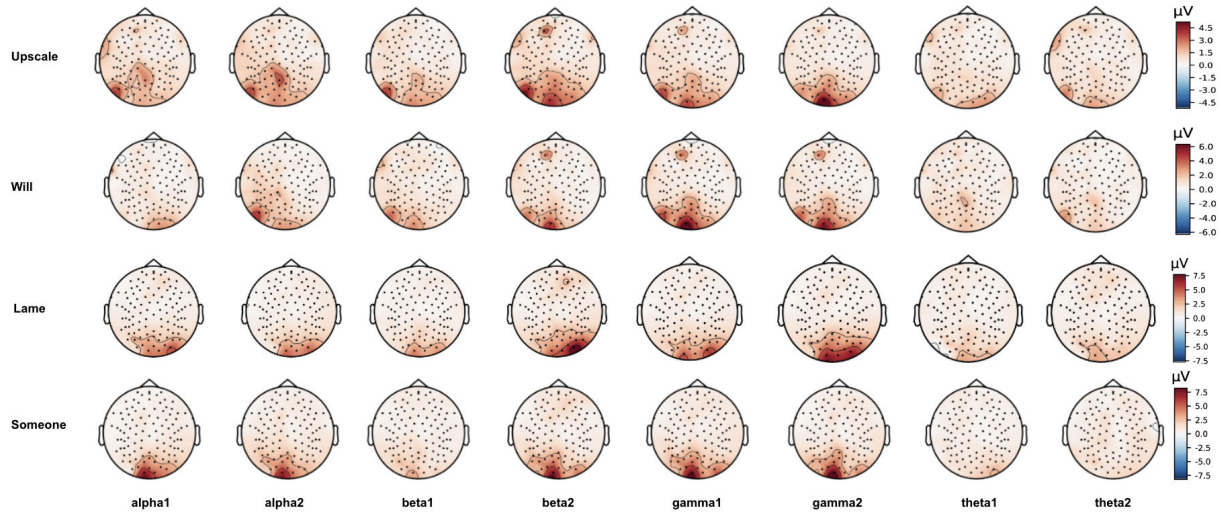


Figure 10.8: Positive and Negative word brain topologies (Sentiment Analysis)

with the comprehension of speech [313].

Similar to Figure 10.5,10.6, we can observe beta2, gamma1, and gamma2 frequency bands have the most powerful signals for all words. In Figure 10.8, activity in Wernicke’s area is seen most visibly in the beta2, gamma1, and gamma2 bands for the words ‘Upscale’ and ‘Will’. For the word ‘Upscale,’ we also saw activity around Broca’s area for alpha1, alpha2, beta1, beta2, theta1, and theta2 bands. An interesting observation is that for the negative words, ‘Lame’ and ‘Someone’, we see very low activation in Broca’s and Wernicke’s areas. Instead, we see most activity in the occipital lobes and slightly over the inferior parietal lobes. The occipital lobes are noted as the visual processing area of the brain and are associated with memory formation, face recognition, distance, and depth interpretation, and visuospatial perception [381]. The inferior parietal lobes are generally found to be key actors in visuospatial attention and semantic memory [321].

Limitations Since we propose a new task of exploring the relationship between EEG and language, we believe there are several limitations that can be focused on in future work.

- The dataset may not be large enough. Due to the difficulty and time-consumption of collecting human-related data (in addition, to privacy concerns), there are few publicly available datasets that have EEG recordings with corresponding natural language. When compared to other mature tasks (i.e., image classification, object detection, etc), datasets that have a combination of EEG signals and different modalities are rare. In the future, we would like to collect more data on EEG signals with natural language to enhance innovation in this direction.
- The computational architecture, the MTAM model, is relatively straightforward. We agree the dual-encoder architecture is one of the standard paradigms in multimodal learning. Since our target is to explore the connectivity and relationship between EEG and language, we used a straightforward paradigm. Our model’s architecture may be less complex compared to others in different tasks, such as image-text pre-training. However, we purposely avoid complicating the model’s structure due to the size of the training data. We noticed when adding more layers of complexity, the model was more prone to overfitting.

- The literature lacks available published baselines. Since the task is new, there are not enough published works that provide comparable baselines. We understand that the comparison is important, so we implemented several baselines by ourselves, including MLP, Bi-LSTM, Transformer, and ResNet, to provide more convincing judgment and support future work in this area.

10.5 Conclusion

In this study, we explore the relationship between EEG and language. We propose MTAM, a Multimodal Transformer Alignment Model, to observe coordinated representations between the two modalities and employ the transformed representations for downstream applications. Our method achieves state-of-the-art performance on sentiment analysis and relation detection tasks on two public datasets, ZuCo and K-EmoCon. Furthermore, we carry out a comprehensive study to analyze the connectivity and alignment between EEG and language. We observe that the transformed features show less randomness and sparsity. The word-level language-EEG alignment clearly demonstrates the importance of the explored connectivity. We also provide brain topologies as an intuitive understanding of the corresponding activity regions in the brain, which could build the empirical neuropsychological basis for understanding the relationship between EEG and language through computational models.

Chapter 11

Clinical Retrieval System for Cardiovascular Magnetic Resonance Imaging

Self-supervised learning is crucial for clinical imaging applications, given the lack of explicit labels in healthcare. However, unlike ECG and EEG in Chapters 9, 10, conventional approaches that rely on precise vision-language alignment are not always feasible in complex clinical imaging modalities, such as cardiac magnetic resonance (CMR). CMR provides a comprehensive visualization of cardiac anatomy, physiology, and microstructure, making it challenging to interpret. Additionally, CMR reports require synthesizing information from sequences of images and different views, resulting in potentially weak alignment between the study and diagnosis report pair. To overcome these challenges, we propose **CMRformer**, a multimodal learning framework to jointly learn sequences of CMR images and associated cardiologist’s reports. Moreover, one of the major obstacles to improving CMR study is the lack of large, publicly available datasets. To bridge this gap, we collected a large **CMR dataset**, which consists of 13,787 studies from clinical cases. By utilizing our proposed CMRformer and our collected dataset, we achieve remarkable performance in real-world clinical tasks, such as CMR image retrieval and diagnosis report retrieval. Furthermore, the learned representations are evaluated to be practically helpful for downstream applications, such as disease classification. Our work could potentially expedite progress in the CMR study and lead to more accurate and effective diagnosis and treatment. [351] has been implemented by Cleveland Clinic for clinical trials.

11.1 Introduction

The application of deep learning to clinical imaging is a highly researched field given the extensive potential to alleviate overburdened providers through automation [173], improve medical care through standardization [306], and accelerate research through high knowledge discovery [79]. However, many current applications are hindered by the lack of annotated data imposed by high costs [229] and privacy regulations [130]. Recent innovations in self-supervised learning where using pretext tasks or implied knowledge have provided a method for which to reduce the dependence on large annotated datasets. Many proposed frameworks are constrained to a single domain, such as image or text [22, 25, 53, 155]. Yet, this would ignore the natural association between clinical images and written radiology reports containing expert interpretations.

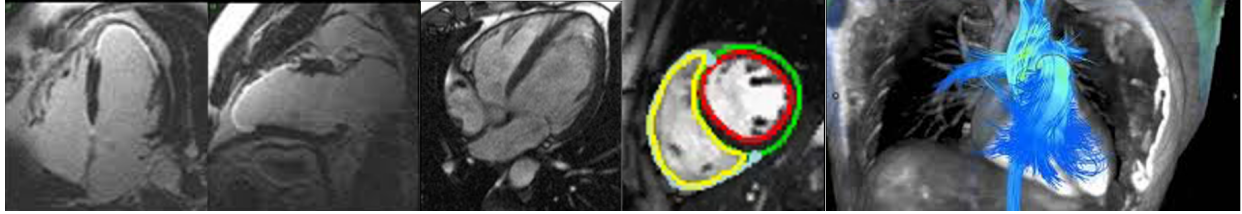


Figure 11.1: Examples of Cardiac magnetic resonance (CMR) images. CMR comprises hundreds of images of differing views and image types, the large majority of which do not contain an indication of pathology. The corresponding report is an analysis of all these images, resulting in difficulty in aligning images with text reports.

Recent developments in contrastive image-text pretraining [367] leverage the natural alignment between image and text pairs to provide co-supervision for each domain and achieved good performance on the natural image-text pairs collected from Internet [62, 93, 120, 212, 241, 242, 248, 251]. However, clinical imaging has unique properties not often seen in natural images. Cardiac magnetic resonance imaging (CMR) is one such example. CMR allows users to visualize the 3D cardiac anatomy and function in an unlimited number of views (although standardized to a few American Heart Association-specified views [408]). CMR studies are able to visualize the morphology, motion, tissue characteristics, and even tissue perfusion within a single study. Each type of image has different characteristics, which make them sensitive to different pathophysiologies. As such, the associated radiology report incorporates findings that describe both individual images and findings that synthesize from multiple image types and views.

Existing image-text pretraining in the natural image domain assumes significant alignment between a single image and a single piece of text. For instance, the natural image MS-COCO dataset [258] uses crowd-sourced short blurbs to describe each image. This simplistic formatting is reflected in the current application of image-text pretraining in clinical imaging domains. Previous works within the clinical imaging field have mostly focused on chest X-ray [189, 198, 485], where the radiology reports are often of limited length, and the image domain is limited to only one or two views of the subject. Furthermore, despite the relatively high resolution of chest X-ray images, the saliency area is often fairly prominent in each image. In contrast, it is not straightforward to incorporate CMR images into current vision-language frameworks, given that only a few images within possibly thousands contained in a CMR study have visible pathology. For example, a ventricular septal defect is a serious morphological abnormality that often requires invasive surgery. However, the defect may be visible in only one or two slices among dozens acquired. Explicitly aligning the written interpretation with individual images is difficult, given that interpretation involves synthesizing images from the whole study, leading to poor alignment between individual images and text.

To address the issue encountered in CMR studies, we propose **CMRformer**, a multimodal learning framework, to jointly learn visual features from CMR images and textual features from radiology reports. Specifically, the CMR data are structured as a sequence of images in video format. Our contribution can be summarized as follows:

- We propose **CMRformer**, a multimodal learning framework that learns visual features from CMR images and textual features from text reports in a joint manner to address the issue of weak alignment between CMR image sequences and their corresponding diagnosis textual reports. The framework is designed to learn from the entire CMR study without the need for

manual identification of specific images relevant to particular diseases.

- The learned embeddings hold immense potential for practical clinical applications, such as the retrieval of CMR studies or radiology reports, searching and retrieving specific information from vast amounts of data in a more efficient manner.
- As the existing literature falls short in providing a large public dataset for CMR, we take the initiative to gather a comprehensive dataset consisting of 13,787 studies derived from actual clinical cases. We also collect and label a Cardiomyopathies dataset for the downstream classification task, which included 1,939 Cardiomyopathies studies and was labeled by clinical experts.

Generalizable Insights about Machine Learning in the Context of Healthcare

In this chapter, we address three fundamental problems existing for clinical CMR imaging learning: (1) the lack of adequate volume of data for individual clinical tasks as well as access to expert labels; (2) the weak alignment between CMR images and the associated reports, which is the bottleneck for multimodal learning; and (3) the lack of functional model which can account for the weak alignment of CMR data to learn useful embeddings for downstream clinical applications. Our work addresses the problems above by:

- We collect a large, single-site CMR dataset consisting of 13,787 studies derived from actual clinical cases. We also collect and label a Cardiomyopathies dataset, with 1,939 studies for the downstream disease classification task.
- We propose CMRformer, a multimodal learning framework that enables the learning of weak alignments between CMR images and corresponding doctor’s reports. The potential applications of this framework are many-fold and far-reaching, including applications in content-based information retrieval, clinical decision support, and healthcare operations. We believe that it represents a significant step forward in the field of medical image analysis and clinical decision-making for CMR study.
- Although this framework is used on CMR data in our study, we expect it can be generalized across several clinical imaging modalities and other healthcare data, including echocardiography, computed tomography, and multi-omic data for the more robust and generalizable application of deep learning to the healthcare domain. Our framework facilitates the extraction of valuable insights from disparate sources of medical data, empowering clinicians with the ability to make more informed and accurate diagnoses and treatment decisions. Ultimately, the widespread adoption of this framework has the potential to significantly improve patient treatment.

11.2 Related Work

Medical Multimodal Learning in Image-Text Setting Medical multimodal learning in the image-text setting focuses on learning the alignment between medical images and accompanying text using a contrastive image-text learning framework. [556] proposed ConVIRT, a contrastive image and text self-supervised learning framework similar to simCLR for chest X-rays, which exceeded the supervised end-to-end method with only 1% of the training data. GLoRIA [181]

introduced a cross-attention layer in order to learn localized similarities in both image and word subdomains. Similarly, LoVT [302] leveraged a projector for localized representations. Recently, [490] proposed to leverage prior knowledge (Unified Medical Language System [40]) as distant supervision to the contrastive learning process. [519] proposed combining contrastive learning with mask language modeling to train ClinicalBERT. [501] explored various data augmentation strategies to improve data efficiency in the clinical realm. However, these methods highly depend on strong alignment between image and text pairs.

Multimodal Learning in Video-Text Setting In video-language pretraining, most clips are not semantically well aligned with their corresponding text [288, 290]. For example, “the basketball player makes a game-winning shot” may have several seconds of additional gameplay and celebration for context. [513] varied the lengths of the video clips and enforced overlapping clips to drive increased similarity between closely related clips, which work for instructional videos or video captioning where there is still a strong assumption of alignment between the action and the text. Alternatively, [26, 44] identified that single frames within videos contribute vast amounts of information. Specifically, [44] leveraged an image-based embedding with a self-attention mechanism to identify the most informative frame for any specific piece of text. [26] proposed a space-time transformer-based encoder that can take both video and image jointly to learn the importance of spatial and temporal features, minimizing the need for well-aligned video-text pairs.

11.3 Proposed Method

To learn better CMR-report multimodal representations, we proposed **CMRformer**, a multimodal learning framework, to jointly learn visual features from CMR studies and text embeddings from the associated radiologists’ reports, based on [26]. The model architecture is shown in Figure 11.2, which contains a visual encoder to process CMR images, and a text encoder to process text reports. More details are introduced in the following sections.

11.3.1 Model Architecture

Visual encoder The visual encoder processes an image or video clip $X \in \mathbb{R}^{M \times 3 \times H \times W}$, where M is the number of frames (1 for images) with a resolution of $H \times W$. It comprises three main components: (i) the patch embedding layer, (ii) learnable embeddings for positional space, time, and [CLS], and (iii) a stack of 12 space-time attention blocks.

To generate patch embeddings, the patch embedding layer uses a 2D convolutional layer with a kernel and stride size equal to the target patch size $P = 16$, with $d = 768$ output channels (the chosen embedding dimension for the video encoder). The positional space and time embeddings have shapes $M \times d$ and $N \times d$, respectively, where M is the maximum number of input video frames, and N is the maximum number of non-overlapping patches of size P within a frame (196 for a video resolution of 224×224). The [CLS] embedding has shape $1 \times d$. Each space-time attention block includes norm layers, temporal and spatial self-attention layers, and a MLP layer, following the approach described in [26].

To process spatio-temporal patches, the video clip input is divided into non-overlapping patches of size $P \times P$, following the protocol in ViT and Timesformer [36]. The resulting patches

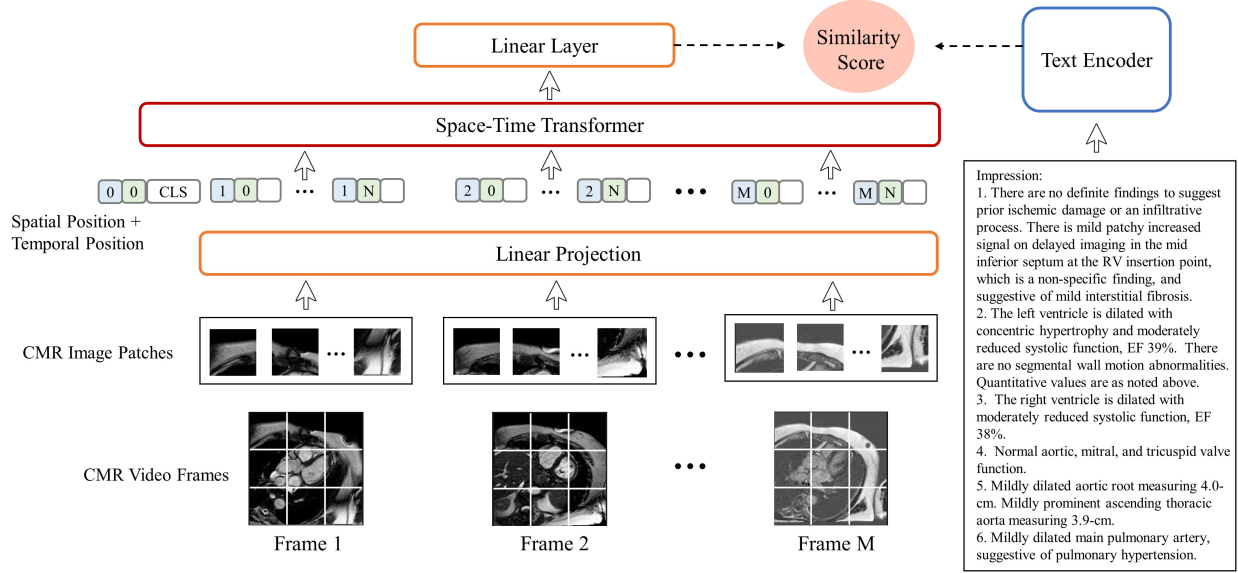


Figure 11.2: The overall architecture of our model, where the visual encoder processes sequences of CMR images and the text encoder processes the text from the “impression” section of the corresponding reports.

$\mathbf{x} \in \mathbb{R}^{M \times N \times 3 \times P \times P}$, where $N = HW/P^2$, are passed through a 2D convolutional layer, and the output is flattened, generating a sequence of embeddings $\mathbf{z} \in \mathbb{R}^{MN \times D}$ that is fed into the transformer. The size of the embeddings, D , depends on the number of kernels in the convolutional layer.

To account for the temporal and spatial position of the patches, learned temporal and spatial positional embeddings, $\mathbf{E}^s \in \mathbb{R}^{N \times D}$ and $\mathbf{E}^t \in \mathbb{R}^{M \times D}$, are added to each input token as follows:

$$\mathbf{z}^{(0)}_{p,m} = \mathbf{z}_{p,m} + \mathbf{E}^s_p + \mathbf{E}^t_m, \quad (11.1)$$

In this equation, all patches within the same frame m (but different spatial locations) receive the same temporal positional embedding \mathbf{E}^t_m , and all patches in the same spatial location (but different frames) receive the same spatial positional embedding \mathbf{E}^s_p . This approach enables the model to identify the temporal and spatial position of each patch. Additionally, a learned [CLS] token [86] is concatenated at the beginning of the sequence to produce the final visual embedding output of the transformer.

The video sequence is processed by a stack of space-time transformer blocks. We introduce a slight modification to the Divided Space-Time attention method proposed by [36], replacing the residual connection between the block input and the temporal attention output with a residual connection between the block input and the spatial attention output. Each block applies temporal self-attention and then spatial self-attention sequentially to the output of the previous block. Finally, the video clip embedding is derived from the [CLS] token of the final block.

Text encoder The text encoder component of our architecture is responsible for processing a sequence of tokenized words and producing a meaningful encoding that captures the semantic content of the input text. To achieve this, we employ a multi-layer bidirectional transformer encoder, which has demonstrated remarkable performance in a wide range of natural language processing

tasks [86, 368]. Specifically, we instantiate the text encoder using the `distilbert-base-uncased` model [402], which is a variant of BERT [86] that has been optimized for efficiency by reducing the number of layers by a factor of 2 and removing the token-type embeddings and pooler components.

At a high level, the text encoder processes the tokenized input sequence by iteratively transforming the embeddings of each token based on its context within the sentence, using a series of self-attention and feed-forward layers. The self-attention mechanism enables the model to attend to different parts of the input sequence when processing each token, while the feed-forward layers allow for nonlinear transformations of the learned representations. Moreover, the bidirectional nature of the encoder allows the model to incorporate information from both past and future tokens in the sequence, resulting in a more comprehensive encoding of the input text.

To obtain the final text encoding, we extract the output of the special [CLS] token in the final layer of the text encoder. This token is specifically designed to provide a summary representation of the input sequence that can be used for downstream tasks such as text classification or information retrieval. The text encoder component plays a crucial role in our architecture by enabling the model to incorporate textual information that complements the visual information encoded by the video encoder.

Projection In order to establish a meaningful association between textual and visual information, it is imperative to first ensure that they are represented in a common feature space. To achieve this, both the textual and visual encodings are projected onto a shared dimension using separate linear layers. Subsequently, the similarity between these projected embeddings is computed by taking their dot product. This approach effectively enables the alignment of heterogeneous modalities, namely textual and visual, and facilitates their comparison in a manner that can be meaningfully interpreted by downstream tasks.

Efficiency The employed model incorporates independent dual encoder pathways, as seen in the MIL-NCE [288] and MMV networks [9], which necessitate a mere dot product computation between the video and text embeddings for establishing meaningful associations between the two modalities. The aforementioned design choice confers upon the model the advantage of a retrieval inference of trivial computational cost, as it can be indexed and efficiently queried using fast approximate nearest neighbor search methods, making it amenable to scaling for very large-scale retrieval tasks at inference time. Specifically, for a given target gallery consisting of v videos and t text queries, the retrieval complexity of our model is $O(t + v)$. By contrast, the ClipBERT [233] model, which adopts a single encoder for both text and video inputs, exhibits a significantly higher retrieval complexity of $O(tv)$, as every possible text-video combination needs to be inputted into the model. Other retrieval methods, such as MoEE [289] and MMT [117], which are based on expert models, also incorporate dual encoder pathways. However, they require query-conditioned weights to calculate similarity scores for each expert, which results in higher computational complexity.

11.3.2 Learning Objectives

Training loss We adopt the approach introduced in [545] for a retrieval-based setting, where pairs of text and video data points in a batch are considered positive matches, while all other pairwise combinations in the batch are considered as negative samples. To facilitate this, we minimize the

sum of two losses, namely video-to-text and text-to-video, given by:

$$L_{v2t} = -\frac{1}{B} \sum_i \log \frac{\exp(x_i^\top y_i / \sigma)}{\sum_{j=1}^B \exp(x_i^\top y_j / \sigma)}, \quad L_{t2v} = -\frac{1}{B} \sum_i \log \frac{\exp(y_i^\top x_i / \sigma)}{\sum_{j=1}^B \exp(y_i^\top x_j / \sigma)} \quad (11.2)$$

Here, x_i and y_j correspond to the normalized embeddings of the i -th video and j -th text, respectively, in a batch of size B , while σ denotes the temperature parameter. The video-to-text loss, L_{v2t} , computes the negative log probability of each video embedding matching its corresponding text embedding, relative to all other text embeddings in the batch, whereas the text-to-video loss, L_{t2v} , computes the negative log probability of each text embedding matching its corresponding video embedding, relative to all other video embeddings in the batch. By minimizing these losses, the model learns to generate embeddings that maximize the similarity between the corresponding video-text pairs and minimize the similarity between non-corresponding pairs.

Frame sampling To capture the temporal information in a video, we divide it into M segments with equal duration, where each segment contains L/M frames. During training, we apply a uniform frame sampling strategy to obtain one frame from each segment. This approach, which is similar to TSN [477] and GST [276], ensures that the model can learn to recognize and capture the salient features across different time segments of the video. At inference time, we adopt a more fine-grained sampling approach by extracting the i -th frame from each segment, where i is determined by a predefined stride S . This results in an array of video embeddings $\mathbf{v} = [v_0, v_S, v_{2S}, \dots, v_{(M-1)S}]$, each of which captures a distinct temporal segment of the video. We then compute the mean of these embeddings, which provides a compact representation of the entire video while preserving the temporal information. This approach allows the model to capture both the short-term and long-term temporal dynamics of the video and can effectively encode the information needed for downstream tasks such as video classification and retrieval. By sampling frames at different intervals during training and testing, our model can learn to recognize temporal patterns at different scales, resulting in robust and accurate representations of the video content.

11.4 Our CMR Dataset

Due to no large publicly available CMR dataset suitable for our study, we collect a **CMR dataset** by ourselves. Our dataset contains CMR images and cardiologists’ text reports of patients who underwent a CMR exam at **Anonymous institution** in both inpatient and outpatient settings between 2008 and 2022. The total size of the initial cohort is 41,936 CMR studies, including a variety of indications according to standard clinical practices. The dataset is collected under a wide range of protocols and machines that evolved with changing clinical standards. We introduce two datasets in our study, one is the **CMR dataset**, which is used for training our CMRformer model and the retrieval task, and the other one is the **Cardiomyopathies dataset**, which is used in the downstream classification task. The **CMR dataset** will be introduced in this section, and the **Cardiomyopathies dataset** is introduced in Section 11.5.3 in the experiment section.

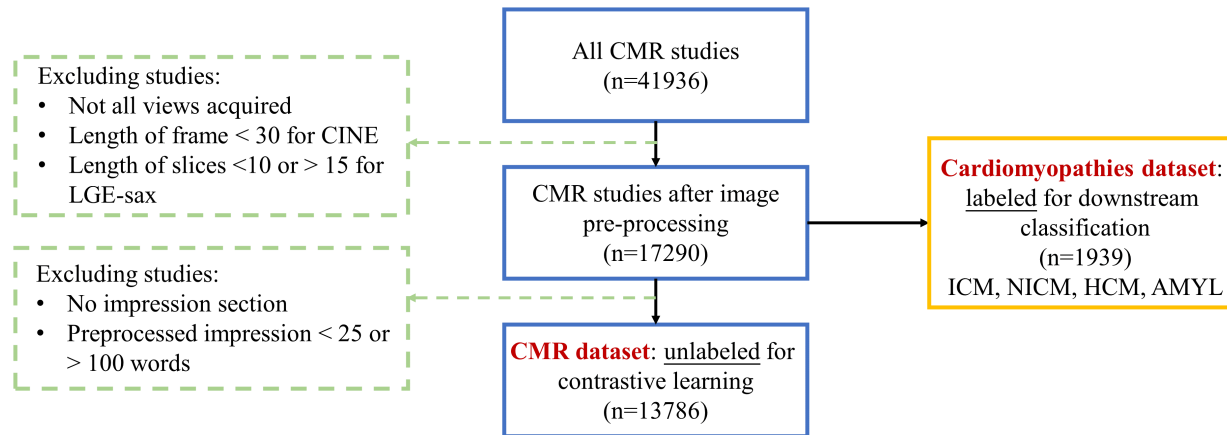


Figure 11.3: Data preprocessing pipeline of our CMR dataset.

11.4.1 Specific Characteristics of the CMR Data

CMR studies differ from chest X-rays and cardiac computed tomography in terms of data type. While the latter two provide single-plane or 3D views of the heart, CMR studies comprise hundreds of 2D images from various angles and image types, with some images being static while others need to be combined to form short video clips. Each image type and view convey unique information. To address this complexity, we use image metadata and established CMR interpretation standards to construct a dictionary that utilizes the “series” DICOM header field, enabling us to specify the cardiac view and image type accurately.

11.4.2 Data Preprocessing and Filtering

The CMR dataset preprocessing pipeline is shown in Figure 11.3, which includes the inclusion/exclusion criteria. The vast majority of cases were collected by Phillips 1.5T Achieva scanner or 3.0T Ingenia scanner. We identified studies that comprised a standard CMR exam targeting ventricular disease.

CMR image types The two image types included in this study are: (1) **CINE**, which are high quality, 2D image clips to capture motion in a single slice of the heart; and (2) **LGE**, which standards for late gadolinium enhancement, a static 2D image which captures tissue viability. Multiple images of a single type are acquired at different slice locations to capture the whole volume of the heart.

CMR image views We target four standard CMR views, including (1) 4 chamber long axis (**lax**), which aims at looking at all 4 chambers of the heart in one view; (2) short axis (**sax**), which aims at visualizing the ventricles; (3) 2 chamber long axis (**2ch**), which aims at left heart visualization; and (4) 3 chamber long axis (**3ch**), which aims at aortic outflow track visualization.

In this chapter, we define the different settings as IMAGE-TYPE_{view} , where **IMAGE-TYPE** is the type of the image, and *view* is the view of the image. The studies without these scans were typically abbreviated study protocols targeting either aortic or valvular disease.

CMR Text Reports The radiology report is an important medico-legal document that contains multiple information about an imaging study to the referring clinician [201]. As such, the report

Table 11.1: Statistics of length of impression sections from text reports.

Text Length	Count	Percentage
20-30	728	5.3%
30-40	2445	17.7%
40-50	2926	21.2%
50-60	2666	19.3%
60-70	2078	15.1%
70-80	1425	10.3%
80-90	929	6.7%
90-100	589	4.3%

Table 11.2: Statistics of the number of images of each study.

Number of Images	Count	Percentage
0-200	61	0.4%
200-300	24	0.2%
300-400	1166	8.5%
400-500	12054	87.4%
500-600	126	0.9%
600-700	79	0.6%
700-800	162	1.2%
> 800	114	0.8%

often contains technical information about the exam, the clinical history of the patient, a general list of imaging biomarkers and other findings, and summary findings split into the *technique*, *indications*, *findings*, and *impressions* section respectively. Much of this information is extraneous to the specific study, making it difficult to learn from, i.e., tabular imaging measurements in the findings section. Therefore, we target the “**impression**” section, which contains a summary of the key findings of each CMR study.

11.4.3 Statistics of the Data

We provide a quantitative analysis of the distribution of the CMR dataset, including the length of the preprocessed texts from the “impression” section, and the number of images from each study. The results are shown in Table 11.1 and Table 11.2, respectively. Based on the tables, we could find that most CMR studies contain 400-500 images, and the length of words within the “impression” section in the corresponding reports is mostly around 30-80. For more demographics of the CMR dataset, the age range is 54.87 ± 15.91 , and within the 13,786 patients, 6,133 are female and 7,653 are male.

11.4.4 Comparison with Existing CMR Datasets

We compare our dataset with several existing public CMR datasets with greater than 100 included studies, including Automated Cardiac Diagnosis Challenge (ACDC) [35], Kaggle 2nd Annual Data Science Bowl Cardiac Challenge (DSB-CC) [145], and Statistical Atlases and Computational Modeling of the Heart (STACOM) [107]. In Table 11.3, it can be observed that our dataset consists of a notably greater number of studies as compared to the datasets currently available. Furthermore, our dataset is unique in that it is directly paired with radiologist-interpreted reports.

ACDC dataset The Automated Cardiac Diagnosis Challenge (ACDC) dataset [35] is a public dataset comprising 150 clinical CMRs acquired at the University Hospital of Dijon, France on either a 1.5T Siemens Area and 3.0T Siemens Trio scanner. The dataset includes only CINE_{sax} images. We used the pathology labels included within the dataset, which include myocardial infarction with systolic heart failure, dilated cardiomyopathy, HCM, abnormal right ventricle, and normal. Classes

Table 11.3: Comparison with existing CMR datasets.

Source	Studies	Image Types	Labels
ACDC	150	Cine	segmentation
DSB-CC	1,140	Cine	end-systolic and end-diastolic volumes
STACOM	<200	varies (mostly Cine)	varies (mostly segmentation)
Ours	13,786/1,939	Cine, LGE	radiology reports/cardiomyopathy diagnosis

are evenly distributed (30 in each class), and the dataset has pre-determined training (100) and test (50) sets.

DSB-CC dataset The 2nd Annual Data Bowl by Kaggle and Booz Allen Hamilton [145] included 1,140 CMR studies. The dataset was limited to only CINE_{sax} images and did not include any disease labels or segmentation labels. Instead, the dataset provided only numeric biomarker measurements, which typically require physician segmentation to acquire. This limits the inherent value of this dataset for pretraining purposes.

Comparison with UK Biobank The UK Biobank contains a large, semi-public CMR dataset representing UK citizens being prospectively followed [338, 339]. Although significantly larger than our data, the UK Biobank is not a good representation of clinical imaging taken within the standard of care. Rather, it is a prospective registry of UK citizens with no specific disease focus and, therefore, is biased heavily toward healthy individuals. The large subset (5000) of the study has been heavily analyzed for imaging biomarkers, and the images can be connected to a multitude of other follow-up data. However, the studies are not interpreted by a radiologist. Furthermore, the CMR studies are collected at four sites with a purposely designed protocol and single-model MR machine making its generalizability to general practice questionable.

11.5 Experiments

11.5.1 Experimental Setting

CMR image types and views In this chapter, we define the different settings as IMAGE-TYPE_{view}, where IMAGE-TYPE is the type of the image, and *view* is the view of the image. Multiple views of the same image type are joined with “-”. For example, CINE_{lax-sax} refers to the long-axis and short-axis view of CINE. An example of video constructed using CINE_{lax-sax} + LGE_{lax-sax-2ch-3ch} is shown in Figure 11.4.

Tasks We include two tasks in the experiments: (1) Retrieval, which includes text-to-video retrieval and video-to-text retrieval. Text-to-video retrieval means retrieving relevant CMR sequences based on a given textual report. Video-to-text retrieval means retrieving relevant textual reports based on a given CMR sequence. (2) Classification, which uses the embeddings from the visual encoder to carry out disease classification on the labeled datasets, including our Cardiomyopathies dataset and public ACDC dataset.

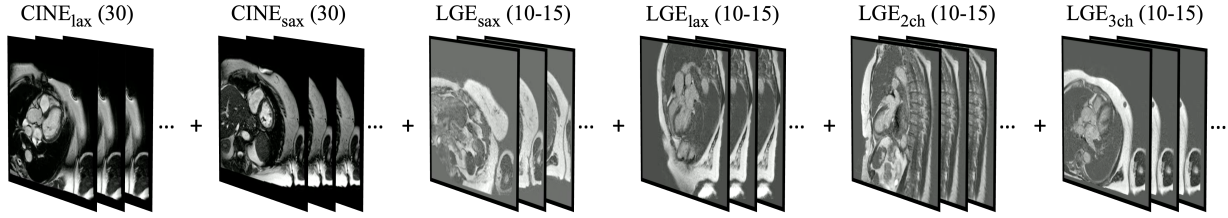


Figure 11.4: Example of CMR image sequences constructed by $CINE_{lax-sax} + LGE_{lax-sax-2ch-3ch}$, where (\cdot) represents the number of images of each type-view combination. For $CINE_{lax-sax}$, each frame represents the time dimension. For LGE_{sax} , each frame corresponds to the depth dimension, and for $LGE_{lax-2ch-3ch}$, each image is duplicated to be consistent with LGE_{sax} .

Evaluation metrics For the retrieval task, we report the recall at K ($R@K$) metric, $K = \{5, 10, 50\}$, where $RSUM$ is defined as the sum of recall metrics at $K = \{5, 10, 50\}$ of both video and text retrieval tasks. For the classification task, we use the standard classification accuracy (Acc), AUC , and $F1$ score as evaluation metrics.

Training details For data splitting of the CMR dataset, we use the 80%/20% split, resulting in 11,028 studies as the training set, and 2,758 studies as the testing set. The model is pretrained on WebVid-2M dataset [26], which contains 2.5M video-text pairs. The detail of model parameters is shown in Table 11.4. Our model is trained on $4 \times$ NVIDIA A100 GPUs.

Table 11.4: Model parameters in the experiments.

Hyperparameter	Value	video params	Value
n_gpu	4	model	SpaceTimeTransformer
optimizer	Adam	arch_config	base_patch16_224
lr	$3e-5$	num_frames	{1, 4, 8, 16, 32, 64}
loss	NormSoftmaxLoss	pretrained	true
epoch	100	time_init	zeros
batch_size	16	text params	Value
extraction_res	256	model	distilbert-base-uncased
input_res	224	pretrained	true
stride	1		

11.5.2 Experimental Results on the Retrieval Task

Learned representations showed better performance than zero-shot results Table 11.5 shows the retrieval results. CMRformer demonstrates remarkable retrieval performance across a wide range of CMR data formats and all metrics, as shown in Table 11.5. Moreover, our model outperforms the zero-shot setting, which uses a model trained solely on WebVid-2M without our CMR dataset. This comparison highlights the effectiveness of our learning approach in improving the model’s performance. Overall, we find that $CINE_{lax-sax} + LGE_{lax-sax-2ch-3ch}$ exhibits the best performance compared to other CMR image type/view combinations.

Table 11.5: Experimental results for retrieval experiments. (·) represents the number of input frames. Zero-Shot evaluation was done using $CINE_{lax-sax} + LGE_{lax-sax-2ch-3ch}$.

Method	Text-to-Video Retrieval			Video-to-Text Retrieval			RSUM
	R@5	R@10	R@50	R@5	R@10	R@50	
Zero-shot (16)	0.3	0.4	1.8	0.2	0.4	1.5	4.6
$CINE_{sax}$ (8)	9.4	15.0	38.6	9.2	14.9	39.1	126.2
$CINE_{lax-sax}$ (8)	13.9	21.1	45.2	13.3	19.9	44.3	157.7
$LGE_{lax-sax-2ch-3ch}$ (8)	14.1	22.3	50.3	14.2	22.3	50.8	174.1
$CINE_{lax-sax} + LGE_{lax-sax}$ (16)	16.4	23.9	54.0	15.4	23.9	54.6	188.1
$CINE_{lax-sax} + LGE_{lax-sax-2ch-3ch}$ (1)	6.3	9.7	27.0	6.3	9.6	27.6	86.7
$CINE_{lax-sax} + LGE_{lax-sax-2ch-3ch}$ (4)	14.5	21.8	46.7	14.0	21.8	45.3	164.0
$CINE_{lax-sax} + LGE_{lax-sax-2ch-3ch}$ (8)	14.8	23.7	51.0	14.4	23.4	51.1	178.5
$CINE_{lax-sax} + LGE_{lax-sax-2ch-3ch}$ (16)	17.9	25.9	53.1	17.3	26.0	54.1	194.3
$CINE_{lax-sax} + LGE_{lax-sax-2ch-3ch}$ (32)	17.7	26.5	55.3	17.8	26.1	56.2	199.8
$CINE_{lax-sax} + LGE_{lax-sax-2ch-3ch}$ (64)	18.5	28.1	56.3	18.1	27.5	56.4	204.8

More types/views contribute better performance The performance of single image types, namely $CINE_{lax-sax}$ and $LGE_{lax-sax-2ch-3ch}$, is observed to be lower in comparison to that of multiple image types/views. CINE offers a dynamic view of the heart’s motion over time, while LGE captures the distribution of fibrosis in the heart. When both image types are included in the radiology report, it enhances the semantic alignment between image sequences and text reports. The inclusion of 2ch and 3ch images provides more information about areas surrounding the aortic and mitral valves, respectively, in addition to the information provided by lax and sax. This inclusion of left heart and valvular views enables better differentiation of certain diseases typically diagnosed using CMR, such as hypertrophic obstructive cardiomyopathy.

Increasing the number of CMR images can result in better performance [290] found that the number of input frames is crucial for the performance of retrieval systems. Our study’s findings indicate that increasing the number of input CMR images can enhance retrieval performance. Clinically, having more frames improves the ability to capture minute adverse changes to cardiac function during different points in the cardiac cycle [23]. Our study revealed that the $CINE_{lax-sax} + LGE_{lax-sax-2ch-3ch}$ (64) outperformed the [1,4,8,16,32] settings. However, larger input sizes require more computational resources and higher machine requirements, such as memory.

11.5.3 Experimental Results on Image Classification Task on our Cardiomyopathies Dataset

Cardiomyopathies dataset The Cardiomyopathies dataset is used for studying the clinical utility of CMR for the diagnosis and prognosis of various cardiomyopathies. There are in total of 1,939 studies included in this dataset, where 1,119 studies of ischemic cardiomyopathy (ICM), 268 cardiac amyloidosis (AMYL), 318 hypertrophic cardiomyopathy (HCM), and 1,357 studies of undifferentiated non-ischemic cardiomyopathy (NICM). For more demographics of the Cardiomyopathies dataset, the age range is 57.96 ± 14.92 , and within the 1,939 patients, 671 are female and 1,268 are

Table 11.6: Linear probing results on the Cardiomyopathies dataset for the downstream disease classification task.

Model	NICM			ICM		
	Acc	AUC	F1	Acc	AUC	F1
Zero-shot	0.69	0.69	0.71	0.77	0.62	0.41
SimCLR	0.71	0.71	0.74	0.75	0.62	0.40
CINE _{sax} (8)	0.75	0.75	0.77	0.79	0.71	0.55
CINE _{lax-sax} (8)	0.80	0.80	0.83	0.84	0.76	0.64
LGE _{lax-sax-2ch-3ch} (8)	0.81	0.81	0.82	0.84	0.79	0.67
CINE _{lax-sax} + LGE _{lax-sax-2ch-3ch} (16)	0.82	0.82	0.84	0.84	0.77	0.65
CINE _{lax-sax} + LGE _{lax-sax-2ch-3ch} (64)	0.84	0.84	0.85	0.84	0.79	0.67

Model	AMYL			HCM		
	Acc	AUC	F1	Acc	AUC	F1
Zero-shot	0.93	0.70	0.43	0.90	0.79	0.66
SimCLR	0.93	0.69	0.43	0.94	0.84	0.78
CINE _{sax} (8)	0.93	0.75	0.50	0.96	0.93	0.87
CINE _{lax-sax} (8)	0.96	0.81	0.66	0.98	0.97	0.94
LGE _{lax-sax-2ch-3ch} (8)	0.96	0.84	0.70	0.98	0.98	0.94
CINE _{lax-sax} + LGE _{lax-sax-2ch-3ch} (16)	0.95	0.80	0.63	0.99	0.99	0.97
CINE _{lax-sax} + LGE _{lax-sax-2ch-3ch} (64)	0.97	0.86	0.76	0.99	0.99	0.97

male. The final diagnosis was identified through a chart review of all available clinical data, not just using the radiology reports. Clinical fellows were tasked with identification according to the relevant clinical guidelines. A level 3 board-certified cardiologist reviewed the results for accuracy.

In order to assess the applicability of our trained model to downstream image classification tasks, we utilize the visual embeddings learned from the visual encoder in the CMRformer and performed linear probing. The training and testing sets for our labeled Cardiomyopathies dataset were split into 70% and 30%, respectively. In our Cardiomyopathies dataset, the number of positive and negative samples were 1049 and 890 for NICM, 461 and 1478 for ICM, 161 and 1778 for AMYL, and 268 and 1671 for HCM. The results of the image classification are presented in Table 11.6. It was observed that the model pretrained on WebVid-2M (zero-shot) did not generalize to CMR without fine-tuning on the CMR data. In addition, although SimCLR [58], a pretrained vision model, has been proven useful in other medical imaging modalities, it did not yield appreciably better results. Our CMRformer achieved significantly better results, demonstrating that the visual embeddings learned by our model can also be useful in downstream image classification tasks.

We have observed a correlation between the linear probing performance and the retrieval performance, suggesting that the trained models have learned valuable CMR representations that can be transferred to downstream tasks. Among the four diseases of interest, linear probing achieves the highest performance on HCM and the lowest on ICM. The superior performance on HCM is expected due to the distinctive morphological differences for HCM patients, such as significantly thickened myocardium. In contrast, NICM, ICM, and AMYL are more challenging to classify. Specifically, AMYL is frequently classified as a subset of NICM, with the main differentiation

Table 11.7: Comparison of the image classification results on the ACDC dataset, where AUC is computed in a one-vs-rest manner and both F1 score and AUC are micro-averaged.

Model	Acc	AUC	F1
Zero-shot	0.30	0.59	0.30
SimCLR	0.42	0.82	0.42
Ours (CINE _{sax}) (8)	0.70	0.95	0.70

being that the pathological processes and treatment for AMYL are better established compared to undifferentiated NICM. ICM can also be challenging to diagnose, as there may be a mild disease or focal disease. Furthermore, in clinical practice, the diagnosis of NICM and ICM is often unclear. Many cases of NICM may have characteristics similar to ICM, such as a focal scar in the absence of coronary obstruction, resulting in partially correct misclassification.

11.5.4 Experimental Results on Image Classification Task on ACDC Dataset

To assess the generalizability of our model, we perform additional experiments on the public ACDC dataset, which only includes CINE_{sax} data. To ensure a fair comparison, we compared our model trained on CINE_{sax} with SimCLR [58] and the zero-shot setting. The multi-class classification results are presented in Table 11.7. Our model outperformed SimCLR, demonstrating its superior generalizability. Our CMRformer’s learned embeddings also proved to be more beneficial in downstream image classification tasks compared to zero-shot results. In order to present more intuitive outcomes, we have employed t-SNE [467] to visualize the visual embedding. The findings are demonstrated in Figure 11.5 and Figure 11.6 for zero-shot setting and trained by CMRformer, respectively. Based on the figures, we observe that the visual embeddings of different classes in the zero-shot setting are intermingled. Conversely, the visual embeddings obtained from our CMRformer are more categorically separated, which elucidates why our approach delivers better image classification outcomes.

11.6 Discussion

Different CMR image types and views We conduct extensive experiments to investigate the performance of various types and views of CMR images. Our findings reveals that the performance of single image types, such as CINE_{lax-sax} and LGE_{lax-sax-2ch-3ch}, is lower when compared to multiple image types/views. Furthermore, the inclusion of 2ch and 3ch images provide additional information about the areas surrounding the aortic and mitral valves, respectively, in addition to the information already provided by lax and sax. This additional information allows for better differentiation of diseases, such as hypertrophic obstructive cardiomyopathy.

Generalizability of our approach Generalizability is a crucial consideration, particularly in the clinical field. To evaluate the suitability of our trained model for downstream image classification tasks, we utilize the visual embeddings acquired from the visual encoder in the CMRformer and conducted linear probing. We observe a positive correlation between the linear probing performance

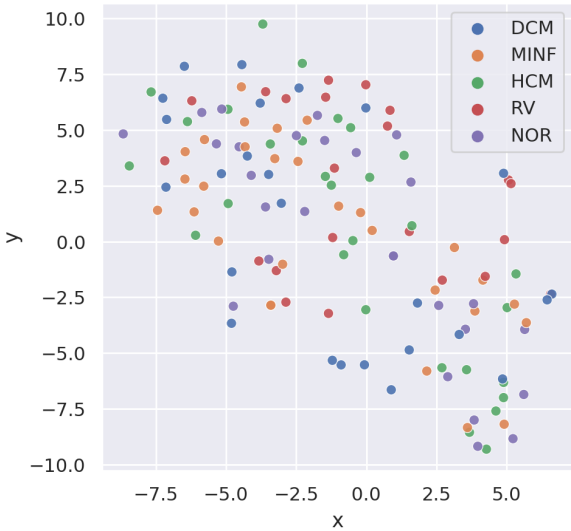


Figure 11.5: t-SNE visualization of zero-shot visual embeddings on the ACDC dataset.

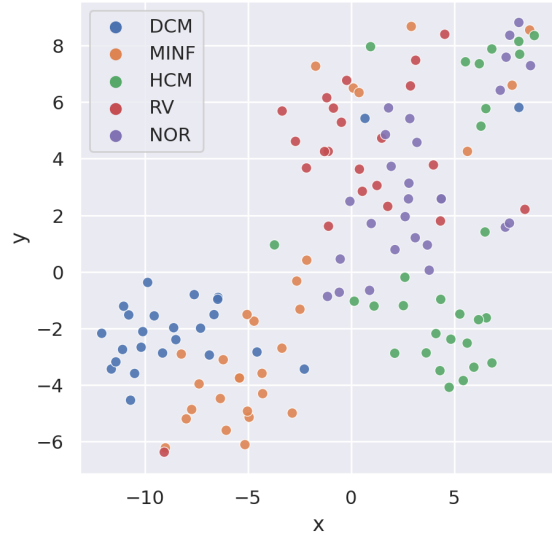


Figure 11.6: t-SNE visualization of learned visual embedding by CMRformer on ACDC.

and the retrieval performance, indicating that our trained models have learned valuable CMR representations that can be transferred to downstream tasks. Additionally, we perform further experiments on the public ACDC dataset and demonstrated that our approach’s generalizability is significantly better than the baseline methods.

Advancement of video-text setting Previous studies [181, 556] have focused on static 2D images, but the true value of clinical imaging lies in 3D, 4D, and sometimes even 5D data. While various data fusion strategies [442] exist to combine data from multiple independent frames, they are not easily adaptable to multi-modal pretraining. Our work proposes a method to learn from complex structured clinical image sequences and associated reports without requiring significant preprocessing. Instead of training individual encoders for each view and image type, the entire study can be processed simultaneously. Furthermore, this approach provides a straightforward way to incorporate other image types into the training process.

Difficulties in learning CMR There are three key factors that make interpreting CMR challenging. First, it requires synthesizing information from both a single frame, such as identifying focal scar on LGE, and motion from a series of frames, such as identifying contractile dysfunction on CINE. Second, CMR patients often have multiple co-morbidities, which contribute to difficulties in identifying the clear cause for clinical symptoms. This ambiguity can lead to variability in interpretation, further misaligning image-text pairs. Finally, there are practical barriers reducing access to CMR, leading to greater than 10-fold less volume compared to other popular clinical modalities. All these issues combined make CMR one of the most difficult settings for the application of machine learning methodologies.

Limitations Despite considerable efforts to organize the data, the data volume pales in comparison to data from other domains. The complexity of the CMR data, consisting of hundreds of images from diverse angles and types, and collected using different scanners, may render the

current model inadequate or insufficiently effective in capturing and processing all the valuable information contained in the CMR image sequences. Furthermore, the limited access to public data makes it challenging to evaluate the model’s generalizability comprehensively. Therefore, further experimentation using multi-center and multi-disease datasets is preferred if more public data becomes available.

11.7 Conclusions

In this chapter, we proposed the first CMR multimodal vision-language contrastive learning framework, which enables the acquisition of CMR representations accompanied by associated cardiologist’s reports. Our contributions are:

- We collect a large, single-site CMR dataset consisting of 13,787 studies derived from clinical cases.
- We propose CMRformer, a multimodal learning framework that enables the learning of weak alignments between CMR images and corresponding doctor’s reports.
- Our model achieved 18.3% performance improvement against the baselines.

Through leveraging this framework, the acquired representations exhibit potential utility in diverse clinical contexts, ranging from the creation of robust retrieval systems to the advancement of disease classification. Our work lays the foundation for future investigations exploring the integration of multimodal learning approaches in medical imaging, which may lead to more accurate diagnoses and improved patient outcomes.

Part V

Conclusions and Future Directions

Chapter 12

Conclusions

12.1 Summary of Contributions

In the current data-driven era, Multimodal Intelligence has become a pivotal concept in the realm of artificial intelligence. This approach leverages data from various modalities, including text, visuals, and audio, to exhibit intelligent behaviors that are more aligned with human-like intelligence. Unlike traditional unimodal techniques that rely on a single data stream, multimodal AI integrates diverse data sources, resulting in more comprehensive and nuanced representations.

This thesis demonstrates that by enhancing the modeling of patterns across different modalities, we can achieve a more effective utilization of the unique modality equivalence learned through abstract multimodal representations. This improved modeling can lead to advancements in cross-modal applications, increasing the robustness of multimodal models under distribution shifts and enhancing their generalization abilities. Consequently, this thesis aims to advance the field of multimodal AI by focusing on the enhancement of alignment, robustness, and generalizability, ultimately leading to the development of more sophisticated and efficient multimodal AI systems.

Below we summarize our contributions from three perspectives: (1) algorithms, (2) datasets and benchmarks, and (3) applications.

Algorithms This thesis has focused on the following algorithmic contributions:

- Multimodal alignment [347, 357, 361]: We explore establishing rich semantic connections between language and image/video data, with a focus on **MSMO** task. By aligning the semantic content of language with visual elements, the resulting models can possess a more nuanced understanding of the underlying concepts.
- Interpretability [361, 363]: We delve into the application of Optimal Transport-based approaches to learn cross-domain alignment, enabling models to provide interpretable explanations of their multimodal reasoning process.
- Language grounding in robot learning [355]: This research aims to develop techniques for learning executable plans from visual observations by incorporating latent language encoding. Models are trained to understand and interpret visual cues while leveraging the rich semantic information encoded in language.
- Retrieval-augmented Multimodal LLM [353]: We develop a retrieval-augmented Multimodal LLM model, which is capable of recognizing and providing knowledgeable answers in

real-world entity-centric VQA.

- ECG-to-text generation [349]: We bridge the gap by transferring the knowledge of LLMs to clinical ECG for diagnosis report generation and zero-shot disease detection.
- Connection between human language and brain signals [146]: We explore the relationship and dependency between EEG and human language to reveal the inner connection.
- ECG-encoding [358]: We encode ECG as images and adopt a vision-language learning paradigm to jointly learn vision-language alignment between encoded ECG images and ECG diagnosis reports. Encoding ECG into images can result in an efficient ECG retrieval system, which can be highly practical and useful in clinical applications.
- Clinical retrieval system for Cardiovascular Magnetic Resonance (CMR) Imaging [351]: We design a retrieval system that can automatically match the input signal to the most similar records in the database. This functionality can significantly aid in diagnosing diseases and reduce physicians' workload.
- Improve robustness through Optimal Transport [362, 574, 575]: We adopt Optimal Transport to improve the model's robustness performance through data augmentation via Wasserstein Geodesic perturbation.

Datasets and Benchmark To provide higher-quality datasets for the community and build benchmarks to provide useful findings based on the literature, this thesis has proposed the following datasets and benchmarks:

- Robustness evaluation benchmark of multimodal models [363]: We develop comprehensive evaluation metrics and methodologies to assess the robustness of multimodal models. By simulating distribution shifts and measuring the model's performance under different scenarios, we can gain a deeper understanding of the model's adaptability and identify potential vulnerabilities.
- New MSMO dataset [357]: We propose a new dataset named MMSum to solve the problems within existing MSMO datasets, such as insufficient maintenance, data inaccessibility, limited size, and categorization, etc., spanning 17 principal categories and 170 subcategories.
- New Livestream video dataset [347]: We introduce a new large dataset of Livestream videos, which contains 11,285 Livestream videos with a total duration of 15,038.4 hours.
- New CMR dataset [351]: The existing work falls short in providing a large CMR dataset, we take the initiative to gather a comprehensive dataset consisting of 13,787 studies derived from actual clinical cases.
- New entity-centric VQA dataset [353]: We have developed the SnapNTell dataset, distinct from traditional VQA datasets as (1) It encompasses a wide range of categorized entities, each represented by images and explicitly named in the answers; (2) It features QA pairs that require extensive knowledge for accurate responses. The dataset is organized into 22 major categories, containing 7,568 unique entities in total.

Applications Throughout this thesis, we have delved into a variety of compelling applications of multimodal AI, spanning diverse fields such as multimedia understanding, robotics learning, and healthcare. To provide a clearer overview of the application areas covered by each piece of work

included in this thesis and during the graduate study, we present Table 12.1, which categorizes the works based on their respective topics.

Table 12.1: Work in topics

	Model/Algorithm	Dataset/Benchmark	Multimedia	Application Robotics	Healthcare	Venue
Alignment	SCCS LiveSeg MMSum	LiveSeg MMSum Entity6K	SCCS LiveSeg MMSum Entity6K			ACL Findings 2023 [361] WACV 2023 [347] CVPR 2024 [357] Under Review [348]
Robustness	MMRobustness Cardiac-MT Interp-OT GeoECG	MMRobustness	Interp-OT		Cardiac-MT GeoECG	DMLR 2024 [363] ICASSP 2023 [362] ICML 2023 [574] PMLR MLHC 2022 [575]
Generalization	SUM+APM ECG-LLM MTAM CMRformer ECG-Encoding SnapNTell	CMRformer SnapNTell	SnapNTell	SUM+APM	ECG-LLM MTAM CMRformer ECG-Encoding	NAACL 2024 [355] EACL Findings 2023 [349] EMNLP Findings 2023 [146] ICML 2023 Workshop [351] ML4H 2023 [358] Under review [353]

12.2 Broader Impact

The work presented in this thesis has already shown significant impact, which we categorize into 1) Academic impact, 2) Impact on the research community through code release, and 3) Application impact.

Application Impact

- [347] has been patented by Adobe for production on Behance Livestream [197].
- [351] has been implemented by Cleveland Clinic for clinical trials.

Academic Impact The work presented here has been previously published in top-tier outlets in different venues, as in Table 12.1, especially:

- [363] was accepted as the very **first** paper of the Journal of Data-centric Machine Learning Research (DMLR).
- [357] was accepted as **Poster (Highlight)** in CVPR 2024, which is **Top 11.9%** among all accepted papers.
- [355] was accepted as a **spotlight** for the ICML 2023 Workshop on Interactive Learning with Implicit Human Feedback.

Impact Through Code Release To enable reproducibility and extendibility, we have publicly released the source code for the algorithms presented in this thesis. As of January 28, 2024:

- [363] has 30 stars, 18 clones, 173 viewers, and 17 citations (as of March 24, 2024).
- [357] has 24 stars, 16 clones, and 424 viewers (as of March 24, 2024).
- [267] has 141 citations.

12.3 Future Directions

12.3.1 Follow-up Work

As more field sciences are incorporating multimodal data, the need for scalable, efficient, and interpretable multimodal learning algorithms will only increase. Below, we outline some future research directions, following the work introduced in this thesis.

Robustness study in a unified setting In our current study, all evaluated multimodal models are pretrained from web-collected data, which likely contains multiple biases and stereotypes, e.g., w.r.t. gender, race, occupation, etc. This is particularly harmful when using large language models or large multimodal models. An important research direction is to study the robustness and fairness of those models in a unified setting. Furthermore, there’s a need for more precise evaluation metrics. Existing metrics may not fully capture the alignment of semantic content across different modalities. For instance, there’s an urgent need for metrics that can quantitatively evaluate the quality of text-to-image generations, taking into account various aspects such as accuracy, diversity, consistency, preference, and more.

Improving robustness through data augmentation In this thesis, we’ve identified the limitations of existing models in handling diverse and challenging datasets, particularly in the presence of visual corruptions and textual perturbations. Enhancing the robustness of these models represents a promising direction for future research. This task is notably challenging and remains largely unresolved. For instance, unimodal data augmentation techniques like Mixup can be easily applied without constraints for unimodal data augmentation. However, multimodal augmentation presents a unique challenge. Given the need for alignment between different modalities, such as in an image-text pair, the augmented data must maintain semantic coherence across both domains. This highlights the challenge of creating methods to quantitatively evaluate the coherence between different modalities.

Physical capabilities of embodied agents In our research on robotics, we concentrate on abstract, high-level actions as described by language instruction without addressing low-level controls. This approach may restrict the effectiveness of the learned policies and their adaptability in complex and dynamic settings. An interesting future direction could involve considering the physical capabilities of embodied agents by developing universal low-level controllers for different morphologies. However, creating these real-world scenarios, particularly for an entire home environment, would pose significant challenges. It might be more feasible to begin with simpler settings and then progress to more complex situations.

Robust embodied policy learning under ambiguous human instructions Ambiguity in human instructions can present obstacles to human-robot interaction. In everyday situations, human instructions are often abstract and open to interpretation, which can compromise the consistency of performance in learning embodied policies. A potential area for future research could be exploring the robustness of robots’ performance when confronted with ambiguous human directives. Following this, approaches to improve this robustness could be devised and put forward.

12.3.2 Broader Discussion

In the above section, we have discussed some specific follow-up work, following the research in this thesis. In this section, we would like to discuss more general topics and challenges in the current multimodal learning literature.

Model architecture and design Multimodal models, released by various organizations worldwide, exhibit distinct characteristics compared to large language models. While the pros and cons of encoder-only, decoder-only, and encoder-decoder architectures have been extensively studied in language modeling, the debate continues regarding the superiority of dual-encoders versus fusion-encoders in multimodal architectures. Additionally, the choice of loss objectives, primarily focused on learning cross-domain alignment, is still under exploration to determine their effectiveness across different data types. Unlike language modeling, where a clear pipeline and learning steps exist, the training of multimodal models appears to lack a defined structure, leading to some randomness in model architecture and design.

In addition, the introduction of multimodal LLMs, which use a modality encoder to map data from other modalities into the language token space and leverage the pretrained language model’s ability for multimodal reasoning, has demonstrated impressive performance in recent studies. This raises intriguing questions about the most meaningful types of learning and whether training multimodal models from scratch is necessary.

Some pretrained models, including CLIP and CoCa, have not fully disclosed details about their pretraining data, techniques, and other specifics. Moreover, models that do offer publicly available weights present a high cost for complete retraining from scratch for an in-depth ablation study. Such a study could determine the effectiveness of different model architecture designs, loss objectives, and augmentation techniques in the learning process. It would benefit the community to collaborate and develop an optimal strategy for multimodal pertaining, which could significantly reduce the duplication of effort and energy consumption, allowing the community to focus on more important issues and contribute to environmental conservation.

Data quality and efficiency Current trends suggest that larger number of model parameters and extensive training data can lead to improved performance. However, the emphasis on increasing model size is being challenged by studies indicating that data quality may be more critical. While existing literature often assumes the availability of abundant data, implying that larger datasets enable models to learn more about the world, our practical experience may contradict this. We have observed that randomly collected data points from the internet, such as image-text pairs, may not always be accurate. For instance, the titles of images might have weak semantic alignment, leading to performance bottlenecks in models trained on such datasets, regardless of their size.

Some research has shown that fine-tuning or distilling models on smaller but higher-quality datasets can significantly enhance performance compared to training on much larger but lower-quality datasets. This raises the question of how to design high-quality datasets tailored to specific domains or application areas, emphasizing the need for a strategic approach to dataset curation.

The data collection process is indeed time-consuming and requires a wealth of experience. To enhance the generalizability of models, it is crucial to gather diverse and representative samples for training. However, determining the type and quantity of data needed for effective training presents significant challenges. Future research could focus on examining the representativeness of data

and its impact on model performance. This could pave the way for more efficient model training, minimizing the efforts wasted on data collection and organization.

Quantitative metrics Quantitative metrics are indeed crucial for multimodal research, where the evaluation of model performance across different domains and modalities can be challenging. Unlike language modeling, where established metrics like ROUGE, BLEU, CIDEr, BertScore, and so on, are commonly used, multimodal research still lacks universally accepted quantitative metrics.

For measuring similarity between different domains, existing methods such as cosine similarity, Euclidean distance, and Wasserstein distance are primarily used in the embedding space to compute the similarity between transformed embeddings. However, these metrics may not adequately capture the semantic correlation across domains, highlighting the need for more refined measures that can better represent cross-domain semantic relationships.

Moreover, with the growing interest in generative applications like image, video, text, and music generation, defining metrics to quantitatively evaluate the quality of generated content is becoming increasingly important. Current approaches often rely on human evaluation, which can be subjective and less reproducible. Developing quantitative metrics for assessing the quality of generated content is a promising direction that could enhance the objectivity and reproducibility of evaluations in multimodal research.

Meaningful real-world tasks Exploring more meaningful real-world tasks, particularly those that have not been addressed by traditional models, could be promising directions. This can be approached in two ways. First, we can focus on data types or characteristics that differ from conventional data. For example, our work with Livestream data has revealed significant differences from traditional video data. Livestream data is generally noisier and more random in both visual and audio aspects. These unique characteristics, which may have been overlooked in previous studies, hold great potential for application, as standard approaches may not be effective.

The second direction involves tackling tasks that are new and have not been addressed in the literature. For instance, developing clinical retrieval systems for doctors, which go beyond simple prediction or classification tasks, could significantly improve patient treatment. This direction necessitates researching use cases in real-world applications to identify genuine needs that current studies may have overlooked or underestimated, such as the applications in agriculture and so on.

Unified multimodal systems The abundance of multimodal data in the world, including images, videos, audio, sensor signals, smells, etc, presents a significant opportunity for research. While current studies often focus on two modalities due to computational constraints and data collection challenges, the field of multimodal research is evolving towards more comprehensive systems. These advanced systems could aim to process information from multiple modalities and generate appropriate responses to input, moving beyond the limitations of current systems that primarily operate in a few popular languages.

The development of such united systems requires more sophisticated model designs and information processing techniques. As the field progresses, these systems could potentially extend to multilingual capabilities, enabling them to cater to a broader range of languages and cultural contexts. This trend towards more integrated and versatile multimodal systems holds great promise for real-world applications, where the ability to understand and respond to diverse forms of data is increasingly crucial.

Bibliography

- [1] How to measure brain activity in people. ([document](#)), 10.1
- [2] Movie dialog corpus: A metadata-rich collection of fictional conversations from raw movie scripts. 4.4, 4.3
- [3] Household robots market share, size, trends, industry analysis report. 2021. ([document](#)), 1.3
- [4] Sathyanarayanan N. Aakur and Sudeep Sarkar. A perceptual prediction framework for self supervised event segmentation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1197–1206, 2019. 4.2, 5.2
- [5] Sadiq H. Abdulhussain, Abd. Rahman bin Ramli, M. Iqbal Saripan, Basheera M. Mahmmod, Syed Abdul Rahman Al-Haddad, and Wissam A. Jassim. Methods and challenges in shot boundary detection: A review. *Entropy*, 20, 2018. 4.2
- [6] U. Rajendra Acharya, Shu Lih Oh, Yuki Hagiwara, Jen Hong Tan, Muhammad Adam, Arkadiusz Gertych, and Ru San Tan. A deep convolutional neural network model to classify heartbeats. *Computers in biology and medicine*, 89:389–396, 2017. 9.2
- [7] Michael Ahn et al. Do as i can, not as i say: Grounding language in robotic affordances. *ArXiv*, abs/2204.01691, 2022. 9.1
- [8] Jean-Baptiste Alayrac et al. Flamingo: a visual language model for few-shot learning. *ArXiv*, abs/2204.14198, 2022. 2.1, 6.1, 8.2, 8.4.2, 8.9
- [9] Jean-Baptiste Alayrac, Adrià Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. In *NeurIPS*, 2020. 11.3.1
- [10] Ahmed Aldahdooh, Wassim Hamidouche, and Olivier Déforges. Reveal of vision transformers robustness against adversarial attacks. *ArXiv*, abs/2106.03734, 2021. 6.1, 6.2, 6.2
- [11] Sawsan Alqahtani, Garima Lalwani, Yi Zhang, Salvatore Romeo, and Saab Mansour. Using optimal transport as alignment objective for fine-tuning multilingual contextualized embeddings. In *EMNLP*, 2021. 3.2
- [12] Emily Alsentzer, John R. Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew B. A. McDermott. Publicly available clinical bert embeddings. *ArXiv*, abs/1904.03323, 2019. 9.6
- [13] Salah Al-Zaiti, Lucas Besomi, Zeineb Bouzid, Ziad Faramand, Stephanie O. Frisch, Christian Martin-Gill, Richard E. Gregg, Samir F. Saba, Clifton Callaway, and Ervin Sejdić. Machine learning-based prediction of acute coronary syndrome using only the pre-hospital 12-lead

- electrocardiogram. *Nature Communications*, 11, 2020. [9.2](#)
- [14] Ezra A. Amsterdam, J. Douglas Kirk, David A. Bluemke, Deborah B. Diercks, Michael E. Farkouh, J. Lee Garvey, Michael C Kontos, James McCord, Todd D. Miller, Anthony P Morise, L. Kristin Newby, Frederick L. Ruberg, Kristine Anne Scordo, and Paul D. Thompson. Testing of low-risk patients presenting to the emergency department with chest pain: a scientific statement from the american heart association. *Circulation*, 122 17:1756–76, 2010. [1.1](#)
- [15] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. SPICE: semantic propositional image caption evaluation. *CoRR*, abs/1607.08822, 2016. [7.4.1](#)
- [16] Galen Andrew, R. Arora, Jeff A. Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *ICML*, 2013. [2.2](#), [4.4](#), [4.4](#), [4.4](#), [10.3.3](#), [10.3.3](#), [10.3.3](#)
- [17] Evlampios E. Apostolidis, E. Adamantidou, Alexandros I. Metsai, Vasileios Mezaris, and I. Patras. Ac-sum-gan: Connecting actor-critic and generative adversarial networks for unsupervised video summarization. *IEEE Transactions on Circuits and Systems for Video Technology*, 31:3278–3292, 2021. [4.5](#), [4.9](#)
- [18] Evlampios E. Apostolidis, E. Adamantidou, Alexandros I. Metsai, Vasileios Mezaris, and I. Patras. Video summarization using deep neural networks: A survey. *Proceedings of the IEEE*, 109:1838–1863, 2021. [5.2](#)
- [19] Yash Kumar Atri, Shraman Pramanick, Vikram Goyal, and Tanmoy Chakraborty. See, hear, read: Leveraging multimodality with guided attention for abstractive text summarization. *ArXiv*, abs/2105.09601, 2021. [2.1](#), [3.2](#), [5.2](#), [5.1](#)
- [20] Sandra Avila, Ana Paula Brandão Lopes, Antonio da Luz, and Arnaldo de Albuquerque Araújo. Vsumm: A mechanism designed to produce static video summaries and a novel evaluation method. *Pattern Recognit. Lett.*, 32:56–68, 2011. [4.3](#), [4.3](#), [5.2](#), [5.3.3](#)
- [21] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Yitzhak Gadre, Shiori Sagawa, Jenia Jitsev, Simon Kornblith, Pang Wei Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. Openflamingo: An open-source framework for training large autoregressive vision-language models. *ArXiv*, abs/2308.01390, 2023. [8.5](#)
- [22] Shekoofeh Azizi, Basil Mustafa, Fiona Ryan, Zachary Beaver, Jan Freyberg, Jonathan Deaton, Aaron Loh, Alan Karthikesalingam, Simon Kornblith, and Ting Chen. Big self-supervised models advance medical image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3478–3488, 2021. [11.1](#)
- [23] Sören J. Backhaus, Georg Metschies, Marcus Billing, Jonas Schmidt-Rimpler, Johannes T. Kowallick, Roman J. Gertz, Tomas Lapinskas, Elisabeth Pieske-Kraigher, Burkert Pieske, and Joachim Lotz. Defining the optimal temporal and spatial resolution for cardiovascular magnetic resonance imaging feature tracking. *Journal of Cardiovascular Magnetic Resonance*, 23(1):1–12, 2021. ISBN: 1532-429X Publisher: BioMed Central. [11.5.2](#)
- [24] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2015. [2.1](#), [3.2](#), [5.2](#)
- [25] Wenjia Bai, Chen Chen, Giacomo Tarroni, Jinming Duan, Florian Guitton, Steffen E. Petersen,

- Yike Guo, Paul M. Matthews, and Daniel Rueckert. Self-Supervised Learning for Cardiac MR Image Segmentation by Anatomical Position Prediction. pages 541–549. Springer International Publishing, 2019. [11.1](#)
- [26] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1708–1718, 2021. [11.2](#), [11.3](#), [11.3.1](#), [11.5.1](#)
- [27] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *IEE Evaluation@ACL*, 2005. [7.4.1](#), [9.5.1](#)
- [28] Lanqing Bao, Jieli Qiu, Hao Tang, Wei-Long Zheng, and Bao-Liang Lu. Investigating sex differences in classification of five emotions from eeg and eye movement signals. *EMBC*, pages 6746–6749, 2019. [4.4](#), [9.2](#), [10.2](#)
- [29] Khayrul Bashar. Ecg and eeg based multimodal biometrics for human identification, 10 2018. [10.2](#)
- [30] Marcel C. M. Bastiaansen, Marieke van der Linden, Mariken Ter Keurs, Ton Dijkstra, and Peter Hagoort. Theta responses are involved in lexical-semantic retrieval during language processing. *Journal of Cognitive Neuroscience*, 17:530–541, 03 2005. [10.4.6](#)
- [31] Behnam Behinaein, Anubha Bhatti, Dirk Rodenburg, Paul C. Hungler, and Ali Etemad. A transformer architecture for stress detection from eeg. *2021 International Symposium on Wearable Computers*, 2021. [9.2](#)
- [32] Yonatan Belinkov and Yonatan Bisk. Synthetic and natural noise both break neural machine translation. *ArXiv*, abs/1711.02173, 2018. [6.5](#)
- [33] Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. *ArXiv*, abs/2004.05150, 2020. [5.4.2](#), [5.5](#), [5.5.3](#)
- [34] Ahmed Ben Said, Amr Mohamed, Tarek Elfouly, Khaled Harras, and Z. Jane Wang. Multimodal deep learning approach for joint eeg-emg data compression and classification, 03 2017. [10.2](#)
- [35] Olivier Bernard, Alain Lalande, Clement Zotti, Frederick Cervenansky, Xin Yang, Pheng-Ann Heng, Irem Cetin, Karim Lekadir, Oscar Camara, Miguel Angel Gonzalez Ballester, et al. Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: is the problem solved? *IEEE transactions on medical imaging*, 37(11):2514–2525, 2018. [11.4.4](#), [11.4.4](#)
- [36] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? *arXiv:2102.05095*, 2021. [11.3.1](#), [11.3.1](#)
- [37] Srinadh Bhojanapalli et al. Understanding robustness of transformers for image classification. *ICCV*, pages 10211–10221, 2021. [6.1](#), [6.2](#), [6.2](#)
- [38] Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Yue Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, Nicolas Pinto, and Joseph P. Turian. Experience grounds language. In *EMNLP*, 2020. [1.1](#), [2.1](#), [10.1](#)
- [39] David M. Blei, A. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003. [5.2](#)

- [40] Olivier Bodenreider. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270, 2004. ISBN: 0305-1048 Publisher: Oxford University Press. 11.2
- [41] Robert O Bonow, Douglas L Mann, Douglas P Zipes, and Peter Libby. *Braunwald’s heart disease e-book: A textbook of cardiovascular medicine*. Elsevier Health Sciences, 2011. 9.2
- [42] C.J. Breen, G.P. Kelly, and W.G. Kernohan. Ecg interpretation skill acquisition: A review of learning, teaching and assessment. *Journal of Electrocardiology*, 2019. 9.2
- [43] Tom B. Brown et al. Language models are few-shot learners. *ArXiv*, abs/2005.14165, 2020. 2.1, 6.5, 7.1, 9.1, 10.1
- [44] Shyamal Buch, Cristóbal Eyzaguirre, Adrien Gaidon, Jiajun Wu, Li Fei-Fei, and Juan Carlos Niebles. Revisiting the" video" in video-language understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2917–2927, 2022. 11.2
- [45] Arthur Fender C. Bucker, Luis F. C. Figueredo, Sami Haddadin, Ashish Kapoor, Shuang Ma, Sai Vemprala, and Rogerio Bonatti. Latte: Language trajectory transformer. *ArXiv*, abs/2208.02918, 2022. 7.2
- [46] Mike Cadogan. ECG Lead positioning. 2020. 9.2
- [47] Brandon Castellano. Intelligent scene cut detection and video splitting tool. <https://bcastell.com/projects/PySceneDetect/>, 2021. 3.3.1, 3.5, 4.5, 5.4.2, 5.5.3
- [48] Kai-Wei Chang, He He, Robin Jia, and Sameer Singh. Robustness and adversarial examples in natural language processing. *EMNLP: Tutorial Abstracts*, 2021. 6.1, 6.2, 6.2
- [49] Yingshan Chang, Mridu Baldevraj Narang, Hisami Suzuki, Guihong Cao, Jianfeng Gao, and Yonatan Bisk. Webqa: Multihop and multimodal qa. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16474–16483, 2021. 8.2, 8.3, 8.3.5
- [50] Chao Che, Peiliang Zhang, Min Zhu, Yue Qu, and Bo Jin. Constrained transformer network for ecg signal processing and arrhythmia classification. *BMC Medical Informatics and Decision Making*, 21, 2021. 9.2
- [51] Harr Chen, S. R. K. Branavan, Regina Barzilay, and David R. Karger. Global models of document structure using latent permutations. In *NAACL*, 2009. 5.2
- [52] Jingqiang Chen and Hai Zhuge. Abstractive text-image summarization using multi-modal attentional hierarchical rnn. In *EMNLP*, pages 4046–4056, 2018. 1.1, 3.4, 5.1, 5.5.3
- [53] Liang Chen, Paul Bentley, Kensaku Mori, Kazunari Misawa, Michitaka Fujiwara, and Daniel Rueckert. Self-supervised learning for medical image analysis using image context restoration. *Medical Image Analysis*, 58:101539, 2019. 11.1
- [54] Liqun Chen, Zhe Gan, Y. Cheng, Linjie Li, L. Carin, and Jing jing Liu. Graph optimal transport for cross-domain alignment. *ICML*, abs/2006.14744, 2020. 2.2, 3.2, 3.3.5, 4.4, 4.4, 10.3.3, 10.3.3
- [55] Liqun Chen, Yizhe Zhang, Ruiyi Zhang, Chenyang Tao, Zhe Gan, Haichao Zhang, Bai Li, Dinghan Shen, Changyou Chen, and Lawrence Carin. Improving sequence-to-sequence learning via optimal transport. *ArXiv*, abs/1901.06283, 2019. 3.2, 4.4

- [56] Shixing Chen, Xiaohan Nie, David D. Fan, Dongqing Zhang, Vimal Bhat, and Raffay Hamid. Shot contrastive self-supervised learning for scene boundary detection. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9791–9800, 2021. [4.2](#), [5.2](#)
- [57] Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu. Fine-grained video-text retrieval with hierarchical graph reasoning. *CVPR*, pages 10635–10644, 2020. [2.1](#)
- [58] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *ArXiv*, abs/2002.05709, 2020. [11.5.3](#), [11.5.4](#)
- [59] Xi Chen et al. Pali: A jointly-scaled multilingual language-image model. *ArXiv*, abs/2209.06794, 2022. [2.1](#), [6.4.3](#)
- [60] Xiuying Chen, Shen Gao, Chongyang Tao, Yan Song, Dongyan Zhao, and Rui Yan. Iterative document representation learning towards summarization with polishing. In *EMNLP*, 2018. [3.5](#), [5.5.2](#)
- [61] Yang Chen, Hexiang Hu, Yi Luan, Haitian Sun, Soravit Changpinyo, Alan Ritter, and Ming-Wei Chang. Can pre-trained vision and language models answer visual information-seeking questions? *ArXiv*, abs/2302.11713, 2023. [8.2](#)
- [62] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *ECCV*, 2020. [4.4](#), [6.1](#), [11.1](#)
- [63] Jianpeng Cheng and Mirella Lapata. Neural summarization by extracting sentences and words. In *ACL*, pages 484–494, 2016. [3.4](#), [5.1](#)
- [64] Jaemin Cho, Abhaysinh Zala, and Mohit Bansal. Dall-eval: Probing the reasoning skills and social biases of text-to-image generative transformers. *ArXiv*, abs/2202.04053, 2022. [6.2](#)
- [65] Kyunghyun Cho, Bart van Merriënboer, Caglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *EMNLP*, 2014. [3.3.5](#), [7.3.4](#)
- [66] Freddy Y. Y. Choi. Advances in domain independent linear text segmentation. In *ANLP*, 2000. [5.2](#)
- [67] Jaekeol Choi, Euna Jung, Jangwon Suh, and Wonjong Rhee. Improving bi-encoder document ranking models with two rankers and multi-teacher distillation. *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2192–2196, 07 2021. [10.2](#), [10.3.1](#)
- [68] Aakanksha Chowdhery et al. Palm: Scaling language modeling with pathways. *ArXiv*, abs/2204.02311:240:1–240:113, 2022. [2.1](#), [2.1](#)
- [69] Joon Son Chung. Naver at activitynet challenge 2019 - task b active speaker detection (ava). *ArXiv*, abs/1906.10555, 2019. [4.5](#)
- [70] V. Cohen-Addad, Varun Kanade, Frederik Mallmann-Trenn, and C. Mathieu. Hierarchical clustering: Objective functions and algorithms. In *SODA*, 2018. [4.5](#), [4.6](#)
- [71] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory

- transformer for image captioning. *CVPR*, pages 10575–10584, 2019. [7.3.4](#)
- [72] J. Correia, E. Formisano, G. Valente, L. Hausfeld, B. Jansma, and M. Bonte. Brain-based translation: fmri decoding of spoken words in bilinguals reveals language-independent semantic representations in anterior temporal lobe. *Journal of Neuroscience*, 34:332–338, 12 2013. [10.3.1](#)
- [73] Francesco Croce, Maksym Andriushchenko, Vikash Sehwal, Edoardo DeBenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. *arXiv preprint arXiv:2010.09670*, 2020. [6.2](#)
- [74] Yuchen Cui, Scott Niekum, Abhi Gupta, Vikash Kumar, and Aravind Rajeswaran. Can foundation models perform zero-shot task specification for robot manipulation? *ArXiv*, abs/2204.11134, 2022. [9.1](#)
- [75] Aron Culotta, Andrew McCallum, and Jonathan Betz. Integrating probabilistic extraction models and data mining to discover relations and patterns in text. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 296–303. Association for Computational Linguistics, 2006. [10.4.1](#)
- [76] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26:2292–2300, 2013. [3.3.5](#)
- [77] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Albert Li, Pascale Fung, and Steven C. H. Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *ArXiv*, abs/2305.06500, 2023. [8.5](#), [8.9](#)
- [78] Giannis Daras and Alexandros G Dimakis. Discovering the hidden vocabulary of dalle-2. *arXiv preprint arXiv:2206.00169*, 2022. [6.1](#), [6.2](#)
- [79] Sabyasachi Dash, Sushil Kumar Shakyawar, Mohit Sharma, and Sandeep Kaushik. Big data in healthcare: management, analysis and future prospects. *Journal of Big Data*, 6(1):54, June 2019. [11.1](#)
- [80] Sandra Eliza Fontes De Avila, Ana Paula Brandão Lopes, Antonio da Luz Jr, and Arnaldo de Albuquerque Araújo. Vsumm: A mechanism designed to produce static video summaries and a novel evaluation method. *Pattern Recognition Letters*, 32(1):56–68, 2011. [3.4](#), [5.1](#), [5.4.2](#), [5.4](#), [5.5.3](#)
- [81] Pinar Demetci, Rebecca Santorella, Björn Sandstede, William Stafford Noble, and Ritambhara Singh. Gromov-wasserstein optimal transport to align single-cell multi-omics data. *bioRxiv*, 2020. [2.2](#), [4.4](#), [10.3.3](#)
- [82] Li Deng. Deep learning: from speech recognition to language and multimodal processing. *APSIPA Transactions on Signal and Information Processing*, 5, 2016. [10.1](#)
- [83] Fatma Deniz, Christine Tseng, Leila Wehbe, and Jack L. Gallant. Semantic representations during language comprehension are affected by context. *bioRxiv*, 2021. [10.1](#)
- [84] Michael J. Denkowski and Alon Lavie. Meteor universal: Language specific translation evaluation for any target language. In *WMT@ACL*, 2014. [5.5.2](#), [6.4.2](#), [8.5.1](#)
- [85] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *ArXiv*, abs/2305.14314, 2023. [8.3.3](#), [8.5.1](#)

- [86] J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019. [3.5](#), [4.4](#), [7.1](#), [7.2](#), [7.3.2](#), [7.4.2](#), [9.1](#), [9.6](#), [10.1](#), [10.2](#), [10.4.3](#), [11.3.1](#), [11.3.1](#)
- [87] Vipula Dissanayake, Sachith Seneviratne, Rajib Rana, Elliott Wen, Tharindu Kaluarachchi, and Suranga Nanayakkara. Sigrep: Toward robust wearable emotion recognition with contrastive representation learning. *IEEE Access*, 10:18105–18120, 2022. [10.4](#)
- [88] Josip Djolonga et al. On robustness and transferability of convolutional neural networks. *CVPR*, 2021. [6.1](#), [6.2](#), [6.2](#)
- [89] Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah A. Smith. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *ArXiv*, abs/2002.06305, 2020. [7.2](#)
- [90] Jianfeng Dong, Xirong Li, Chaoxi Xu, Xun Yang, Gang Yang, Xun Wang, and Meng Wang. Dual encoding for video retrieval by text. *IEEE transactions on pattern analysis and machine intelligence*, PP, 2021. [2.1](#)
- [91] Xinshuai Dong, Anh Tuan Luu, Rongrong Ji, and Hong Liu. Towards robustness against natural language word substitutions. *ArXiv*, abs/2107.13541, 2021. [6.1](#), [6.2](#), [6.2](#)
- [92] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929, 2020. [2.2](#), [5.4.3](#), [6.2](#)
- [93] Zi-Yi Dou, Yichong Xu, Zhe Gan, Jianfeng Wang, Shuohang Wang, Lijuan Wang, Chenguang Zhu, Nanyun Peng, Zicheng Liu, and Michael Zeng. An empirical study of training end-to-end vision-and-language transformers. *ArXiv*, abs/2111.02387, 2021. <https://github.com/zdou0830/meter>. [6.1](#), [6.4.1](#), [11.1](#)
- [94] Nathan G. Drenkow, Numair Sani, Ilya Shpitser, and M. Unberath. Robustness in deep learning for computer vision: Mind the gap? *ArXiv*, abs/2112.00639, 2021. [6.1](#), [6.2](#), [6.2](#)
- [95] Yang Du, Yongling Xu, Xiaoan Wang, Li Liu, and Pengcheng Ma. Etst: Eeg transformer for person identification. 04 2022. [10.4.3](#)
- [96] Jiali Duan, Liqun Chen, Son Thai Tran, Jinyu Yang, Yi Xu, Belinda Zeng, and Trishul M. Chilimbi. Multi-modal alignment using representation codebook. *ArXiv*, abs/2203.00048, 2022. [3.1](#), [3.2](#)
- [97] J. Ebrahimi, Daniel Lowd, and Dejing Dou. On adversarial examples for character-level neural machine translation. In *COLING*, 2018. [6.5](#)
- [98] Desmond Elliott, Douwe Kiela, and Angeliki Lazaridou. Multimodal learning and reasoning. In *ACL 2016*, 2016. [10.1](#)
- [99] Nick Erickson, Xingjian Shi, James Sharpnack, and Alexander J. Smola. Multimodal automl for image, text and tabular data. *ACM SIGKDD*, 2022. [1.1](#), [2.1](#)
- [100] Isak Czeresnia Etinger and Alan W. Black. Formality style transfer for noisy, user-generated conversations: Extracting labeled, parallel data from unlabeled corpora. In *EMNLP*, 2019. [6.3.2](#)

- [101] B. Everitt and A. Skrondal. Comprar the cambridge dictionary of statistics | b. s. everitt | 9780521766999 | cambridge university press. 2010. [4.5](#)
- [102] Alexander W. Fang, Gabriel Ilharco, Mitchell Wortsman, Yuhao Wan, Vaishaal Shankar, Achal Dave, and Ludwig Schmidt. Data determines distributional robustness in contrastive language image pre-training (clip). 2022. [6.2](#)
- [103] Javier Fdez, Nicholas Guttenberg, Olaf Witkowski, and Antoine Pasquali. Cross-subject eeg-based emotion recognition through neural networks with stratified normalization. *Frontiers in Neuroscience*, 15, 02 2021. [10.4.3](#)
- [104] Jonathan G. Fiscus, George R. Doddington, John S. Garofolo, and Alvin F. Martin. Nist’s 1998 topic detection and tracking evaluation (tdt2). In *EUROSPEECH*, 1999. [4.5](#)
- [105] Rémi Flamary et al. Pot: Python optimal transport. 2021. [3.2](#), [3.3.5](#)
- [106] Joseph L. Fleiss. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76:378–382, 1971. [8.3.3](#)
- [107] Carissa G Fonseca, Michael Backhaus, David A Bluemke, Randall D Britten, Jae Do Chung, Brett R Cowan, Ivo D Dinov, J Paul Finn, Peter J Hunter, Alan H Kadish, et al. The cardiac atlas project—an imaging database for computational modeling and statistical atlases of the heart. *Bioinformatics*, 27(16):2288–2295, 2011. [11.4.4](#)
- [108] Stanislav Fort. Pixels still beat text: Attacking the openai clip model with text patches and adversarial pixel perturbations. 2021. [6.1](#), [6.2](#)
- [109] Chris Foster, Chad C. Williams, Olave E. Krigolson, and Alona Fyshe. Using eeg to decode semantics during an artificial language learning task. *Brain and Behavior*, 11, 2021. [10.2](#)
- [110] Emily B. Fox. Bayesian nonparametric learning of complex dynamical phenomena. 2009. [4.4](#)
- [111] C. Ailie Fraser, Joy Kim, Hijung Shin, Joel Brandt, and Mira Dontcheva. Temporal segmentation of creative live streams. *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 2020. [4.2](#)
- [112] Daniel Fried, Armen Aghajanyan, Jessy Lin, Sida I. Wang, Eric Wallace, Freda Shi, Ruiqi Zhong, Wen tau Yih, Luke Zettlemoyer, and Mike Lewis. Incoder: A generative model for code infilling and synthesis. *ArXiv*, abs/2204.05999, 2022. [9.1](#)
- [113] Daniel Fried, Jean-Baptiste Alayrac, Phil Blunsom, Chris Dyer, Stephen Clark, and Aida Nematzadeh. Learning to segment actions from observation and narration. In *ACL*, 2020. [4.4](#)
- [114] Xiyang Fu, J. Wang, and Zhenglu Yang. Multi-modal summarization for video-containing documents. *ArXiv*, abs/2009.08018, 2020. [2.1](#), [3.1](#), [3.2](#), [3.4](#), [3.4](#), [3.5](#), [3.5](#), [5.1](#), [5.2](#), [5.1](#), [5.4.3](#), [5.5.3](#)
- [115] Xiyang Fu, Jun Wang, and Zhenglu Yang. Mm-avs: A full-scale dataset for multi-modal summarization. In *NAACL*, 2021. [2.1](#), [3.1](#), [3.2](#), [3.4](#), [3.4](#), [3.5](#), [3.5](#), [5.2](#), [5.2](#), [5.3.3](#), [5.5.3](#)
- [116] Antonino Furnari, Sebastiano Battiato, and Giovanni Maria Farinella. Personal-location-based temporal segmentation of egocentric video for lifelogging applications. *Journal of Visual Communication and Image Representation*, 52:1–12, 2018. [4.2](#)
- [117] Valentin Gabeur, Chen Sun, Karteeek Alahari, and Cordelia Schmid. Multi-modal transformer

- for video retrieval. In *ECCV*, 2020. [11.3.1](#)
- [118] Samir Yitzhak Gadre, Mitchell Wortsman, Gabriel Ilharco, Ludwig Schmidt, and Shuran Song. Clip on wheels: Zero-shot object navigation as object localization and exploration. *ArXiv*, abs/2203.10421, 2022. [7.2](#)
- [119] Yuri Galindo and Fabio A. Faria. Understanding clip robustness. 2021. [6.1](#), [6.2](#)
- [120] Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. Large-scale adversarial training for vision-and-language representation learning. *ArXiv*, abs/2006.06195, 2020. [6.1](#), [11.1](#)
- [121] Zhanning Gao, Le Wang, Qilin Zhang, Zhenxing Niu, Nanning Zheng, and Gang Hua. Video imprint segmentation for temporal action detection in untrimmed videos. In *AAAI*, 2019. [4.2](#)
- [122] Jon Gauthier and Anna A. Ivanova. Does the brain represent words? an evaluation of brain decoding studies of language understanding. *ArXiv*, abs/1806.00591, 2018. [10.3.1](#)
- [123] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *ArXiv*, abs/1811.12231, 2019. [6.2](#), [6.3.1](#)
- [124] Deepti Ghadiyaram, Matt Feiszli, Du Tran, Xueting Yan, Heng Wang, and Dhruv Kumar Mahajan. Large-scale weakly-supervised pre-training for video action recognition. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, abs/1905.00561:12038–12047, 2019. [5.4.3](#)
- [125] John R. Giudicessi, Matthew Schram, J. Martijn Bos, Conner Galloway, Jacqueline Baras Shreibati, Patrick W. Johnson, Rickey E. Carter, Levi W Disrud, Robert B Kleiman, Zach I. Attia, Peter A. Noseworthy, Paul A. Friedman, David E. Albert, and Michael J. Ackerman. Artificial intelligence-enabled assessment of the heart rate corrected qt interval using a mobile electrocardiogram device. *Circulation*, 2021. [9.2](#)
- [126] Goran Glavas, Federico Nanni, and Simone Paolo Ponzetto. Unsupervised text segmentation using semantic relatedness graphs. In **SEMEVAL*, 2016. [5.2](#)
- [127] Karan Goel, Nazneen Rajani, Jesse Vig, Samson Tan, Jason M. Wu, Stephan Zheng, Caiming Xiong, Mohit Bansal, and Christopher R’e. Robustness gym: Unifying the nlp evaluation landscape. In *NAACL*, 2021. [6.1](#), [6.2](#), [6.2](#)
- [128] Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. Multimodal neurons in artificial neural networks. *Distill*, 6(3):e30, 2021. [6.1](#), [6.2](#)
- [129] Yunchao Gong, Qifa Ke, Michael Isard, and Svetlana Lazebnik. A multi-view embedding space for modeling internet images, tags, and their semantics. *International Journal of Computer Vision*, 106:210–233, 2013. [2.2](#), [4.4](#), [10.3.3](#)
- [130] Benjamin M. Good and Andrew I. Su. Crowdsourcing for bioinformatics. *Bioinformatics*, 29(16):1925–1933, 2013. ISBN: 1460-2059 Publisher: Oxford University Press. [11.1](#)
- [131] Nishad Gothoskar, Miguel Lázaro-Gredilla, Abhishek Agarwal, Yasemin Bekiroglu, and Dileep George. Learning a generative model for robot control using visual feedback. *ArXiv*, abs/2003.04474, 2020. [7.2](#)

- [132] Priya Goyal, Quentin Duval, Isaac Seessel, Mathilde Caron, Ishan Misra, Levent Sagun, Armand Joulin, and Piotr Bojanowski. Vision models are more robust and fair when pretrained on uncurated images without supervision. *ArXiv*, abs/2202.08360, 2022. 6.1, 6.2, 6.2
- [133] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. *International Journal of Computer Vision*, 127:398 – 414, 2016. 8.3.5, 8.3
- [134] Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural networks*, 18 5-6:602–10, 2005. 3.3.1, 3.3.1, 3.3.3, 10.4.3
- [135] Yuxian Gu, Robert Tinn, Hao Cheng, Michael R. Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3:1 – 23, 2022. 9.6
- [136] Liangke Gui, Borui Wang, Qiuyuan Huang, Alexander G. Hauptmann, Yonatan Bisk, and Jianfeng Gao. Kat: A knowledge augmented transformer for vision-and-language. In *North American Chapter of the Association for Computational Linguistics*, 2021. 8.2
- [137] Tao Gui et al. Textflint: Unified multilingual robustness evaluation toolkit for natural language processing. In *ACL*, 2021. 6.1, 6.2, 6.2
- [138] Chenfeng Guo and Dongrui Wu. Canonical correlation analysis (cca) based multi-view learning: An overview. *ArXiv*, abs/1907.01693, 2019. 4.4
- [139] Dalu Guo, Chang Xu, and Dacheng Tao. Image-question-answer synergistic network for visual dialog. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10426–10435, 2019. 3.4
- [140] Tanmay Gupta, Ryan Marten, Aniruddha Kembhavi, and Derek Hoiem. Grit: General robust image task benchmark. *ArXiv*, abs/2204.13653, 2022. 6.2
- [141] Michael U Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS*, 2010. 2.1
- [142] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. Realm: Retrieval-augmented language model pre-training. *ArXiv*, abs/2002.08909, 2020. 8.1, 8.2
- [143] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. Creating summaries from user videos. In *ECCV*, 2014. 3.1, 3.5, 4.1, 4.2, 4.3, 4.3, 4.5, 5.2, 5.2, 5.1, 5.2, 5.3.3
- [144] John Hale, Adhiguna Kuncoro, Keith B. Hall, Chris Dyer, and Jonathan Brennan. Text genre and training data size in human-like parsing. In *EMNLP*, 2019. 10.2
- [145] Booz Allen Hamilton. Kaggle 2nd annual data science bowl cardiac challenge. <https://www.kaggle.com/competitions/second-annual-data-science-bowl>, 2015. 11.4.4, 11.4.4
- [146] William Han, Jieli Qiu, Jiacheng Zhu, Mengdi Xu, Douglas Weber, Bo Li, and Ding Zhao. An empirical exploration of cross-domain alignment between language and electroencephalogram. *ArXiv*, abs/2208.06348, 2022. 1.1, 1.2, 1.3, 1.1, 2.1, 4.4, 9.2, 12.1, 12.1

- [147] Darryl Hannan, Akshay Jain, and Mohit Bansal. Manymodalqa: Modality disambiguation and qa over diverse inputs. In *AAAI Conference on Artificial Intelligence*, 2020. [8.2](#), [8.3](#), [8.3.5](#)
- [148] Xiaoshuai Hao, Yi Zhu, Srikar Appalaraju, Aston Zhang, Wanqian Zhang, Boyang Li, and Mu Li. Mixgen: A new multi-modal data augmentation. *ArXiv*, abs/2206.08358, 2022. [6.5](#)
- [149] Li Haopeng, Ke Qihong, Gong Mingming, and Tom Drummond. Progressive video summarization via multimodal self-supervised learning. 2022. [5.2](#)
- [150] Li Haopeng, Ke Qihong, Gong Mingming, and Zhang Rui. Video summarization based on video-text modelling. 2022. [3.1](#)
- [151] J. A. Hartigan and M. Anthony. Wong. A k-means clustering algorithm. 1979. [5.4.2](#), [5.4](#), [5.5.3](#)
- [152] Ahmed Hassanien, Mohamed A. Elgharib, Ahmed A. S. Seleim, Mohamed Hefeeda, and Wojciech Matusik. Large-scale, fast and accurate shot boundary detection through spatio-temporal convolutional neural networks. *ArXiv*, abs/1705.03281, 2017. [4.2](#), [5.2](#)
- [153] Eman Hato and Matheel Emaduldeen Abdulmunem. Fast algorithm for video shot boundary detection using surf features. *2019 2nd Scientific Conference of Computer Sciences (SCCS)*, pages 81–86, 2019. [4.2](#), [5.2](#)
- [154] Bo He, Jun Wang, Jieli Qiu, Trung Bui, Abhinav Shrivastava, and Zhaowen Wang. Align and attend: Multimodal summarization with dual contrastive losses. *ArXiv*, abs/2303.07284, 2023. [2.1](#), [5.2](#)
- [155] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. [11.1](#)
- [156] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CVPR*, pages 770–778, 2016. [3.3.5](#), [4.3](#), [4.4](#), [7.4.3](#), [9.3](#), [10.4.3](#)
- [157] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Lixuan Zhu, Samyak Parajuli, Mike Guo, Dawn Xiaodong Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *ICCV*, 2021. [6.5](#)
- [158] Dan Hendrycks and Thomas G. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *ArXiv*, abs/1903.12261, 2019. [6.2](#), [6.3.1](#)
- [159] Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Xiaodong Song. Pretrained transformers improve out-of-distribution robustness. In *ACL*, 2020. [6.2](#), [7.2](#)
- [160] Dan Hendrycks, Norman Mu, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. *ArXiv*, abs/1912.02781, 2020. [6.5](#)
- [161] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Xiaodong Song. Natural adversarial examples. *CVPR*, 2021. [6.2](#)
- [162] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay,

- Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *NIPS*, pages 1693–1701, 2015. [3.4](#)
- [163] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, 2017. [6.4.2](#), [6.4.3](#)
- [164] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9:1735–1780, 1997. [3.3.3](#), [9.3](#)
- [165] Nora Hollenstein, Maria Barrett, and Lisa Beinborn. Towards best practices for leveraging human language processing signals for natural language processing. In *LINCR*, 2020. [10.2](#)
- [166] Nora Hollenstein, Maria Barrett, Marius Troendle, Francesco Bigiolli, Nicolas Langer, and Ce Zhang. Advancing nlp with cognitive language processing signals. *ArXiv*, abs/1904.02682, 2019. [10.1](#), [10.2](#)
- [167] Nora Hollenstein, Cédric Renggli, Benjamin James Glaus, Maria Barrett, Marius Troendle, Nicolas Langer, and Ce Zhang. Decoding eeg brain activity for multi-modal natural language processing. *Frontiers in Human Neuroscience*, 15, 2021. [9.2](#), [10.2](#), [10.2](#), [10.3.1](#), [10.4.3](#), [10.4.3](#), [10.2](#), [10.4.6](#)
- [168] Nora Hollenstein, Jonathan Rotsztein, Marius Troendle, Andreas Pedroni, Ce Zhang, and Nicolas Langer. Zuco, a simultaneous eeg and eye-tracking resource for natural sentence reading. *Scientific Data*, 5, 12 2018. [10.4.1](#), [10.4.1](#), [10.4.2](#)
- [169] Nora Hollenstein, Marius Troendle, Ce Zhang, and Nicolas Langer. Zuco 2.0: A dataset of physiological recordings during natural reading and annotation. *arXiv:1912.00903 [cs]*, 03 2020. [10.4.1](#), [10.4.1](#), [10.4.2](#)
- [170] Yicong Hong, Qi Wu, Yuankai Qi, Cristian Rodriguez-Opazo, and Stephen Gould. Vln-bert: A recurrent vision-and-language bert for navigation. *CVPR*, pages 1643–1653, 2021. [7.2](#)
- [171] Or Honovich, Roei Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Y. Matias. True: Re-evaluating factual consistency evaluation. In *Workshop on Document-grounded Dialogue and Conversational Question Answering*, 2022. [3.5](#)
- [172] Chiori Hori et al. End-to-end audio visual scene-aware dialog using multimodal attention-based video features. In *ICASSP*, pages 2352–2356, 2019. [1.1](#), [3.4](#), [5.1](#), [5.5.3](#)
- [173] Ahmed Hosny, Chintan Parmar, John Quackenbush, Lawrence H. Schwartz, and Hugo JWL Aerts. Artificial intelligence in radiology. *Nature Reviews Cancer*, 18(8):500–510, 2018. ISBN: 1474-175X Publisher: Nature Publishing Group UK London. [11.1](#)
- [174] Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. In *ACL*, 2018. [7.2](#)
- [175] Renee Y. Hsia, Zachariah Hale, and Jeffrey A. Tabas. A national study of the prevalence of life-threatening diagnoses in patients with chest pain. *JAMA internal medicine*, 176 7:1029–32, 2016. [1.1](#)
- [176] Wan Ting Hsu, Chieh-Kai Lin, Ming-Ying Lee, Kerui Min, Jing Tang, and Min Sun. A unified model for extractive and abstractive summarization using inconsistency loss. *ArXiv*, abs/1805.06266, 2018. [3.4](#)

- [177] Hexiang Hu, Yi Luan, Yang Chen, Urvashi Khandelwal, Mandar Joshi, Kenton Lee, Kristina Toutanova, and Ming-Wei Chang. Open-domain visual entity recognition: Towards recognizing millions of wikipedia entities. *ArXiv*, abs/2302.11154, 2023. [8.2](#)
- [178] Ronghang Hu, Daniel Fried, Anna Rohrbach, Dan Klein, Trevor Darrell, and Kate Saenko. Are you looking? grounding to multiple modalities in vision-and-language navigation. In *ACL*, 2019. [1.1](#), [2.1](#)
- [179] Xiaowei Hu, Zhe Gan, Jianfeng Wang, Zhengyuan Yang, Zicheng Liu, Yumao Lu, and Lijuan Wang. Scaling up vision-language pretraining for image captioning. *CVPR*, 2022. [1.1](#), [2.1](#)
- [180] Shih-Cheng Huang, Anuj Pareek, Roham Zamanian, Imon Banerjee, and Matthew P. Lungren. Multimodal fusion with deep neural networks for leveraging ct imaging and electronic health record: a case-study in pulmonary embolism detection. *Scientific Reports*, 10:22147, 12 2020. [10.2](#)
- [181] Shih-Cheng Huang, Liyue Shen, Matthew P. Lungren, and Serena Yeung. Gloria: A multimodal global-local representation learning framework for label-efficient medical image recognition. In *International Conference on Computer Vision*, pages 3942–3951, 2021. [11.2](#), [11.6](#)
- [182] Wenlong Huang, P. Abbeel, Deepak Pathak, and Igor Mordatch. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In *ICML*, 2022. [7.2](#), [9.1](#)
- [183] Wenlong Huang, F. Xia, Ted Xiao, Harris Chan, Jacky Liang, Peter R. Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Noah Brown, Tomas Jackson, Linda Luu, Sergey Levine, Karol Hausman, and Brian Ichter. Inner monologue: Embodied reasoning through planning with language models. In *Conference on Robot Learning*, 2022. [7.2](#)
- [184] Xun Huang and Serge J. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. *ICCV*, 2017. [6.1](#)
- [185] Yu Huang, Chenzhuang Du, Zihui Xue, Xuanyao Chen, Hang Zhao, and Longbo Huang. What makes multimodal learning better than single (provably). In *NeurIPS*, 2021. [10.1](#)
- [186] Zhihu Huang and Jinsong Leng. Analysis of hu’s moment invariants on image scaling and rotation. *2010 2nd International Conference on Computer Engineering and Technology*, 7:V7–476–V7–480, 2010. [5.4.2](#), [5.5.3](#)
- [187] Lorenz Hübschle-Schneider and Peter Sanders. Parallel weighted random sampling. *ACM Transactions on Mathematical Software (TOMS)*, 2019. [10.4.3](#)
- [188] Drew A. Hudson and Christopher D. Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6693–6702, 2019. [8.3.5](#), [8.3](#)
- [189] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, and Katie Shpanskaya. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 590–597, 2019. Issue: 01. [11.1](#)

- [190] Shruti Jadon and Mahmood Jasim. Video summarization using keyframe extraction and video skimming. *arXiv preprint arXiv:1910.04792*, 2019. [5.4.2](#), [5.4](#), [5.5.3](#)
- [191] Shruti Jadon and Mahmood Jasim. Unsupervised video summarization framework using keyframe extraction and video skimming. In *ICCCA*, pages 140–145, 2020. [3.1](#), [4.1](#), [5.2](#)
- [192] Vidhi Jain, Yixin Lin, Eric Undersander, Yonatan Bisk, and Akshara Rai. Transformers are adaptable task planners. *ArXiv*, abs/2207.02442, 2022. [1.1](#), [2.1](#), [9.1](#)
- [193] Zhong Ji, Kailin Xiong, Yanwei Pang, and Xuelong Li. Video summarization with attention-based encoder–decoder networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(6):1709–1717, 2020. [3.3.3](#)
- [194] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. [4.4](#)
- [195] Hao Jiang and Yadong Mu. Joint video summarization and moment localization by cross-task sample transfer. In *CVPR*, pages 16388–16398, 2022. [5.1](#)
- [196] Yunfan Jiang, Agrim Gupta, Zichen Vincent Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi (Jim) Fan. Vima: General robot manipulation with multimodal prompts. *ArXiv*, abs/2210.03094, 2022. [7.2](#)
- [197] Hailin Jin, Jielin Qiu, Zhaowen Wang, Trung Huu Bui, and Franck Deroncourt. Multimodal unsupervised video temporal segmentation for summarization, November 30 2023. US Patent App. 17/804,656. [1.3](#), [4](#), [12.2](#)
- [198] Alistair EW Johnson, Tom J. Pollard, Seth J. Berkowitz, Nathaniel R. Greenbaum, Matthew P. Lungren, Chih-ying Deng, Roger G. Mark, and Steven Horng. MIMIC-CXR, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317, 2019. ISBN: 2052-4463 Publisher: Nature Publishing Group UK London. [11.1](#)
- [199] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7:535–547, 2017. [8.4.1](#)
- [200] Matthew J. Johnson and Alan S. Willsky. Bayesian nonparametric hidden semi-markov models. *J. Mach. Learn. Res.*, 14:673–701, 2013. [4.4](#), [4.4](#), [4.4](#), [4.4](#)
- [201] Charles E Kahn Jr, Curtis P Langlotz, Elizabeth S Burnside, John A Carrino, David S Channin, David M Hovsepian, and Daniel L Rubin. Toward best practices in radiology reporting. *Radiology*, 252(3):852–856, 2009. [11.4.2](#)
- [202] Hussain Kanafani, Junaid Ahmed Ghauri, Sherzod Hakimov, and Ralph Ewerth. Unsupervised video summarization via multi-source features. *Proceedings of the 2021 International Conference on Multimedia Retrieval*, 2021. [4.2](#)
- [203] Yash Kant et al. Housekeep: Tidying virtual households using commonsense reasoning. *ArXiv*, abs/2205.10712, 2022. [7.2](#), [9.1](#)
- [204] Jared Kaplan, Sam McCandlish, T. J. Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeff Wu, and Dario Amodei. Scaling laws for neural language models. *ArXiv*, abs/2001.08361, 2020. [9.1](#)
- [205] Akbar Karimi, L. Rossi, and Andrea Prati. Aeda: An easier data augmentation technique for

- text classification. In *EMNLP*, 2021. [6.3.2](#)
- [206] Lei Ke, Wenjie Pei, Ruiyu Li, Xiaoyong Shen, and Yu-Wing Tai. Reflective decoding network for image captioning. *ICCV*, pages 8887–8896, 2019. [7.3.1](#)
- [207] M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30:81–93, 1938. [8.5.2](#)
- [208] M. G. Kendall, Alan L. Stuart, and J. Keith Ord. Kendall’s advanced theory of statistics. *Journal of the American Statistical Association*, 90:398, 1995. [8.5.2](#)
- [209] Apoorv Khandelwal, Luca Weihs, Roozbeh Mottaghi, and Aniruddha Kembhavi. Simple but effective: Clip embeddings for embodied ai. *CVPR*, pages 14809–14818, 2022. [7.2](#), [9.1](#)
- [210] Aman Khullar and Udit Arora. Mast: Multimodal abstractive summarization with trimodal hierarchical attention. *arXiv preprint arXiv:2010.08021*, 2020. [1.1](#), [5.1](#)
- [211] Shaan Khurshid, Samuel N. Friedman, Christopher Reeder, Paolo Di Achille, Nathaniel Diamant, Pulkit Singh, Lia X. Harrington, Xin Wang, Mostafa A. Al-Alusi, Gopal Sarma, Andrea S. Foulkes, Patrick T. Ellinor, Christopher D Anderson, Jennifer E. Ho, Anthony A. Philippakis, Puneet Batra, and Steven A. Lubitz. Electrocardiogram-based deep learning and clinical risk factors to predict atrial fibrillation. *Circulation*, 2021. [9.2](#)
- [212] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *ICML*, 2021. <https://github.com/dandelin/ViLT>. [4.4](#), [6.1](#), [6.4.1](#), [6.4.3](#), [11.1](#)
- [213] Serkan Kiranyaz, Turker Ince, Ridha Hamila, and M. Gabbouj. Convolutional neural networks for patient-specific ecg classification. *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2608–2611, 2015. [9.2](#)
- [214] Ryan Kiros, Ruslan Salakhutdinov, and Richard S. Zemel. Multimodal neural language models. In *ICML*, 2014. [10.1](#)
- [215] Johannes Klicpera, Marten Lienen, and Stephan Günnemann. Scalable optimal transport in high dimensions for graph distances, embedding alignment, and more. *ArXiv*, abs/2107.06876, 2021. [3.2](#)
- [216] William Knight. A computer method for calculating kendall’s tau with ungrouped data. *Journal of the American Statistical Association*, 61:436–439, 1966. [8.5.2](#)
- [217] Vikrant Kobra, David S. Doermann, and Christos Faloutsos. Videotrails: representing and visualizing structure in video sequences. In *MULTIMEDIA ’97*, 1997. [4.2](#)
- [218] Philipp Koehn. Statistical machine translation. 2007. [7.3.4](#)
- [219] Jing Yu Koh, Ruslan Salakhutdinov, and Daniel Fried. Grounding language models to images for multimodal inputs and outputs. 2023. [2.1](#)
- [220] Irena Koprinska and Sergio Carrato. Temporal video segmentation: A survey. *Signal Process. Image Commun.*, 16:477–500, 2001. [4.2](#)
- [221] Thomas Kosch, Albrecht Schmidt, Simon Thanheiser, and Lewis L. Chuang. One does not simply RSVP: Mental workload to select speed reading parameters using electroencephalography. *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 04 2020. [10.4.6](#)
- [222] Omri Koshorek, Adir Cohen, Noam Mor, Michael Rotman, and Jonathan Berant. Text

- segmentation as a supervised learning task. In *NAACL*, 2018. [3.5](#), [5.2](#)
- [223] Mateusz Krubiński and Pavel Pecina. Mlask: Multimodal summarization of video-based news articles. In *Findings*, 2023. [5.4.3](#), [5.4.3](#), [5.4.3](#), [5.4.3](#), [5.4.3](#), [5.4.3](#)
- [224] Hilde Kuehne, Alexander Richard, and Juergen Gall. A hybrid rnn-hmm approach for weakly supervised temporal action segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42:765–779, 2020. [4.2](#), [5.2](#)
- [225] Shuhei Kurita and Kyunghyun Cho. Generative language-grounded policy in vision-and-language navigation with bayes’ rule. *ArXiv*, abs/2009.07783, 2020. [7.2](#)
- [226] Tuomas Kynkaanniemi, Tero Karras, Miika Aittala, Timo Aila, and Jaakko Lehtinen. The role of imagenet classes in fréchet inception distance. *ArXiv*, abs/2203.06026, 2022. [6.4.2](#), [6.4.3](#)
- [227] Xiang Lan, Dianwen Ng, linda Qiao, and Mengling Feng. Intra-inter subject self-supervised learning for multivariate cardiac signals. In *AAAI*, 2022. [9.4](#)
- [228] Colin S. Lea, Michael D. Flynn, René Vidal, Austin Reiter, and Gregory Hager. Temporal convolutional networks for action segmentation and detection. In *CVPR*, 2017. [4.2](#), [5.2](#)
- [229] Chonho Lee, Zhaojing Luo, Kee Yuan Ngiam, Meihui Zhang, Kaiping Zheng, Gang Chen, Beng Chin Ooi, and Wei Luen James Yip. Big healthcare data analytics: Challenges and applications. *Handbook of large-scale distributed computing in smart healthcare*, pages 11–41, 2017. ISBN: 3319582798 Publisher: Springer. [11.1](#)
- [230] John Lee, Max Dabagia, Eva L. Dyer, and Christopher J. Rozell. Hierarchical optimal transport for multimodal distribution alignment. *ArXiv*, abs/1906.11768, 2019. [2.2](#), [3.2](#), [4.4](#), [10.3.3](#)
- [231] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. *ArXiv*, abs/1803.08024, 2018. [2.1](#)
- [232] Szer Ming Lee, John H. Xin, and Stephen Westland. Evaluation of image similarity by histogram intersection. *Color Research and Application*, 30:265–274, 2005. [5.4.2](#), [5.5.3](#)
- [233] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. *arXiv preprint arXiv:2102.06183*, 2021. [11.3.1](#)
- [234] Paul Lerner, Olivier Ferret, Camille Guinaudeau, Hervé Le Borgne, Romaric Besançon, José G. Moreno, and Jesús Lovón-Melgarejo. Viquae, a dataset for knowledge-based visual question answering about named entities. *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2022. [8.3](#), [8.3.5](#)
- [235] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *ACL*, 2020. [3.3.4](#), [3.4](#), [3.5](#), [5.4.2](#), [5.5](#), [5.5.3](#), [7.3.2](#), [9.6](#), [10.1](#)
- [236] Haoran Li, Junnan Zhu, Congbo Ma, Jiajun Zhang, and Chengqing Zong. Multi-modal summarization for asynchronous collection of text, image, audio and video. In *EMNLP*, 2017. [2.1](#), [3.2](#), [5.2](#)

- [237] J. Li, Aixin Sun, and Shafiq R. Joty. Segbot: A generic neural text segmentation model with pointer network. In *IJCAI*, 2018. 5.2
- [238] Juncen Li, Robin Jia, He He, and Percy Liang. Delete, retrieve, generate: a simple approach to sentiment and style transfer. In *NAACL*, 2018. 6.3.2
- [239] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *ArXiv*, abs/2301.12597, 2023. 8.5, 8.9
- [240] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *arXiv:2201.12086 [cs]*, 02 2022. 7.1, 7.2
- [241] Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. <https://github.com/salesforce/BLIP>. 6.1, 6.4.1, 6.4.3, 6.4.3, 7.3.1, 7.4.3, 11.1
- [242] Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Deepak Gotmare, Shafiq R. Joty, Caiming Xiong, and Steven C. H. Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *NeurIPS*, 2021. <https://github.com/salesforce/ALBEF>. 6.1, 6.4.1, 11.1
- [243] Linjie Li, Jie Lei, Zhe Gan, and Jingjing Liu. Adversarial vqa: A new benchmark for evaluating the robustness of vqa models. *ICCV*, 2021. 6.2
- [244] Liunian Harold Li et al. Grounded language-image pre-training. *CVPR*, 2021. 6.4.3, 8.4.1
- [245] Mingzhe Li, Xiuying Chen, Shen Gao, Zhangming Chan, Dongyan Zhao, and Rui Yan. Vmsmo: Learning to generate multimodal summary for video-based news articles. *arXiv preprint arXiv:2010.05406*, 2020. 5.1, 5.4.3
- [246] Mu Li and Bao-Liang Lu. Emotion classification based on gamma-band eeg. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference*, 2009:1323–1326, 2009. 10.4.6
- [247] Shuang Li, Xavier Puig, Yilun Du, Clinton Jia Wang, Ekin Akyürek, Antonio Torralba, Jacob Andreas, and Igor Mordatch. Pre-trained language models for interactive decision-making. *ArXiv*, abs/2202.01771, 2022. 7.1, 7.2, 7.4.1, 7.5.1, 7.5, 7.5.3, 9.1
- [248] Wei Li, Can Gao, Guocheng Niu, Xinyan Xiao, Hao Liu, Jiachen Liu, Hua Wu, and Haifeng Wang. Unimo-2: End-to-end unified vision-language grounded learning. *arXiv preprint arXiv:2203.09067*, 2022. 6.1, 11.1
- [249] Xiangpeng Li, Jingkuan Song, Lianli Gao, Xianglong Liu, Wenbing Huang, Xiangnan He, and Chuang Gan. Beyond rnns: Positional self-attention with co-attention for video question answering. In *AAAI*, 2019. 3.4
- [250] Xiujun Li, Chunyuan Li, Qiaolin Xia, Yonatan Bisk, Asli Celikyilmaz, Jianfeng Gao, Noah A. Smith, and Yejin Choi. Robust navigation with language pretraining and stochastic sampling. In *Conference on Empirical Methods in Natural Language Processing*, 2019. 7.2
- [251] Xiujun Li, Xi Yin, Chunyuan Li, Xiaowei Hu, Pengchuan Zhang, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-semantics

- aligned pre-training for vision-language tasks. In *ECCV*, 2020. [6.1](#), [7.3.1](#), [11.1](#)
- [252] J. Liang, Wenlong Huang, F. Xia, Peng Xu, Karol Hausman, Brian Ichter, Peter R. Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. *ArXiv*, abs/2209.07753, 2022. [7.1](#), [7.2](#), [9.1](#)
- [253] Paul Pu Liang, Amir Zadeh, and Louis-Philippe Morency. Foundations and recent trends in multimodal machine learning: Principles, challenges, and open questions. 2022. [2.1](#)
- [254] Yuan-Hong Liao, Xavier Puig, Marko Boben, Antonio Torralba, and Sanja Fidler. Synthesizing environment-aware activities via activity sketches, 06 2019. [7.4.1](#), [7.4.2](#)
- [255] Bingqian Lin, Yi Zhu, Zicong Chen, Xiwen Liang, Jian zhuo Liu, and Xiaodan Liang. Adapt: Vision-language navigation with modality-aligned action prompts. *CVPR*, pages 15375–15385, 2022. [7.2](#)
- [256] Chi-Heng Lin, Mehdi Azabou, and Eva L. Dyer. Making transport more robust and interpretable by moving data through a small number of anchor points. *Proceedings of machine learning research*, 139:6631–6641, 2021. [4.4](#)
- [257] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *ACL 2004*, pages 74–81, 2004. [3.5](#), [5.5.2](#), [6.4.2](#), [7.4.1](#), [8.5.1](#), [9.5.1](#)
- [258] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. [3.5](#), [6.4.1](#), [11.1](#)
- [259] Tal Linzen. How can we accelerate progress towards human-like linguistic generalization? In *ACL*, 2020. [10.1](#)
- [260] Fenglin Liu, Yuanxin Liu, Xuancheng Ren, Xiaodong He, and Xu Sun. Aligning visual regions and textual concepts for semantic-grounded image representations. In *NeurIPS*, 2019. [4.4](#)
- [261] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *ArXiv*, abs/2310.03744, 2023. [8.5](#)
- [262] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *ArXiv*, abs/2304.08485, 2023. [8.3.3](#), [8.5](#)
- [263] Na Liu, Mark Dras, and W. Zhang. Detecting textual adversarial examples based on distributional characteristics of data representations. In *REPLANLP*, 2022. [6.5](#)
- [264] Nanyang Liu, Xian Sun, Hongfeng Yu, Wenkai Zhang, and Guangluan Xu. Multistage fusion with forget gate for multimodal summarization in open-domain videos. In *Conference on Empirical Methods in Natural Language Processing*, 2020. [5.4.3](#)
- [265] Vivian Liu and Lydia B. Chilton. Design guidelines for prompt engineering text-to-image generative models. *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, 2022. [6.4.3](#)
- [266] Wei Liu, Jie-Lin Qiu, Wei-Long Zheng, and Bao-Liang Lu. Multimodal emotion recognition using deep canonical correlation analysis. *ArXiv*, abs/1908.05349, 2019. [4.4](#), [9.2](#), [10.2](#), [10.3.1](#)
- [267] Wei Liu, Jie-Lin Qiu, Wei-Long Zheng, and Bao-Liang Lu. Comparing recognition per-

- formance and robustness of multimodal deep learning models for multimodal emotion recognition. *IEEE Transactions on Cognitive and Developmental Systems*, 14:715–729, 2022. [1.1](#), [1.3](#), [2.1](#), [4.4](#), [9.2](#), [10.2](#), [12.2](#)
- [268] Yang Liu and Mirella Lapata. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China, November 2019. Association for Computational Linguistics. [5.2](#), [5.2](#)
- [269] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692, 2019. [7.1](#), [7.3.2](#), [9.1](#), [9.6](#), [10.1](#)
- [270] Ziquan Liu, Yi Xu, Yuanhong Xu, Qi Qian, Hao Li, Rong Jin, Xiangyang Ji, and Antoni B. Chan. An empirical study on distribution shift robustness from the perspective of pre-training and data augmentation. *ArXiv*, abs/2205.12753, 2022. [6.3](#)
- [271] Ziyi Liu, Jiaqi Zhang, Yongshuai Hou, Xinran Zhang, Ge Li, and Yang Xiang. Machine learning for multimodal electronic health records-based research: Challenges and perspectives, 2021. [5.1](#)
- [272] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Neural Information Processing Systems*, 2019. [7.3.1](#)
- [273] Jiasen Lu, Christopher Clark, Rowan Zellers, Roozbeh Mottaghi, and Aniruddha Kembhavi. Unified-io: A unified model for vision, language, and multi-modal tasks. *ArXiv*, abs/2206.08916, 2022. [6.4.3](#)
- [274] W. Lu, Yiqiang Chen, Jindong Wang, and Xin Qin. Cross-domain activity recognition via substructural optimal transport. *Neurocomputing*, 2021. [3.3.5](#)
- [275] Michal Lukasik, Boris Dadachev, Kishore Papineni, and Gonçalo Simões. Text segmentation by cross segment attention. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Online, November 2020. Association for Computational Linguistics. [3.3.2](#), [3.5](#), [5.2](#), [5.4.3](#)
- [276] Chenxu Luo and Alan Yuille. Grouped spatial-temporal aggregation for efficient action recognition. In *ICCV*, 2019. [11.3.2](#)
- [277] Edward Ma. Nlp augmentation. 2019. <https://github.com/makcedward/nlpaug>. [6.3.2](#)
- [278] Yecheng Jason Ma et al. Vip: Towards universal visual reward and representation via value-implicit pre-training. *ArXiv*, abs/2210.00030, 2022. [7.2](#)
- [279] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, A. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *ACL*, 2011. [6.5](#)
- [280] Kaleel Mahmood, Rigel Mahmood, and Marten van Dijk. On the robustness of vision transformers to adversarial examples. *ICCV*, 2021. [6.1](#), [6.2](#), [6.2](#)
- [281] Arjun Majumdar, Ayush Shrivastava, Stefan Lee, Peter Anderson, Devi Parikh, and Dhruv Batra. Improving vision-and-language navigation with image-text pairs from the web. *ArXiv*,

- abs/2004.14973, 2020. [7.2](#)
- [282] Emanuele La Malfa and Marta Z. Kwiatkowska. The king is naked: on the notion of robustness for natural language processing. In *AAAI*, 2022. [6.1](#), [6.2](#), [6.2](#)
- [283] Anshu Malhotra and Rajni Jindal. Multimodal deep learning based framework for detecting depression and suicidal behaviour by affective analysis of social media posts. *EAI Endorsed Transactions on Pervasive Health and Technology*, 6(21), 2020. [5.1](#)
- [284] Xiaofeng Mao, Gege Qi, Yuefeng Chen, Xiaodan Li, Ranjie Duan, Shaokai Ye, Yuan He, and Hui Xue. Towards robust vision transformer. *ArXiv*, abs/2105.07926, 2021. [6.1](#), [6.2](#), [6.2](#)
- [285] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3190–3199, 2019. [8.2](#), [8.3.5](#), [8.3](#)
- [286] Thomas Mensink, Jasper R. R. Uijlings, Lluís Castrejón, Arushi Goel, Felipe Cadar, Howard Zhou, Fei Sha, Andre F. de Araújo, and Vittorio Ferrari. Encyclopedic vqa: Visual questions about detailed properties of fine-grained categories. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3090–3101, 2023. [8.3](#), [8.3.5](#)
- [287] Claudio Michaelis et al. Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv preprint arXiv:1907.07484*, 2019. [6.3.1](#)
- [288] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*, 2020. [5.4.3](#), [11.2](#), [11.3.1](#)
- [289] Antoine Miech, Ivan Laptev, and Josef Sivic. Learning a text-video embedding from incomplete and heterogeneous data. *arXiv*, 2018. [11.3.1](#)
- [290] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2630–2640, 2019. [5.3.1](#), [5.4.3](#), [11.2](#), [11.5.2](#)
- [291] Rada Mihalcea and Paul Tarau. Textrank: Bringing order into text. In *EMNLP*, 2004. [3.4](#)
- [292] Tomas Mikolov, Wen tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In *NAACL*, 2013. [5.2](#)
- [293] Derek Miller. Leveraging bert for extractive text summarization on lectures. *ArXiv*, abs/1906.04165, 2019. [5.1](#)
- [294] So Yeon Min, Devendra Singh Chaplot, Pradeep Ravikumar, Yonatan Bisk, and Ruslan Salakhutdinov. Film: Following instructions in language with modular methods. *ArXiv*, abs/2110.07342, 2022. [1.1](#), [2.1](#)
- [295] Li Mingzhe, Xiuying Chen, Shen Gao, Zhangming Chan, Dongyan Zhao, and Rui Yan. Vmsmo: Learning to generate multimodal summary for video-based news articles. In *EMNLP*, 2020. [2.1](#), [3.1](#), [3.2](#), [3.3](#), [3.4](#), [3.4](#), [3.5](#), [3.5](#), [5.2](#), [5.1](#), [5.3.3](#), [5.5.2](#)
- [296] Abdalkarim Mohtasib, Gerhard Neumann, and Heriberto Cuayáhuitl. A study on dense and sparse (visual) rewards in robot policy learning. In *TAROS*, 2021. [7.2](#)

- [297] George B. Moody and Roger G. Mark. The impact of the mit-bih arrhythmia database. *IEEE Engineering in Medicine and Biology Magazine*, 20:45–50, 2001. [9.2](#)
- [298] Seungwhan Moon, Andrea Madotto, Zhaojiang Lin, Tushar Nagarajan, Matt Smith, Shashank Jain, Chun-Fu Yeh, Prakash Murugesan, Peyman Heidari, Yue Liu, Kavya Srinet, Babak Damavandi, and Anuj Kumar. Anymal: An efficient and scalable any-modality augmented language model. *ArXiv*, abs/2309.16058, 2023. [8.4.2](#), [8.5.1](#), [8.5.4](#)
- [299] Milad Moradi and Matthias Samwald. Evaluating the robustness of neural language models to input perturbations. In *EMNLP*, 2021. [6.2](#)
- [300] Louis-Philippe Morency and Tadas Baltrusaitis. Multimodal machine learning: Integrating language, vision and speech. In *ACL*, 2017. [10.1](#)
- [301] Markus U. Müller, Nikoo Ekhtiari, Rodrigo M. Almeida, and Christoph Rieke. Super-resolution of multispectral satellite images using convolutional neural networks. *ArXiv*, abs/2002.00580, 2020. [5.5.2](#)
- [302] Philip Müller, Georgios Kaissis, Congyu Zou, and Daniel Rueckert. Radiological reports improve pre-training for localized imaging tasks on chest x-rays. In *Medical Image Computing and Computer Assisted Intervention—MICCAI 2022: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part V*, pages 647–657. Springer, 2022. [11.2](#)
- [303] Alex Murphy, Bernd Bohnet, Ryan T. McDonald, and Uta Noppeney. Decoding part-of-speech from human eeg signals. In *ACL*, 2022. [10.2](#)
- [304] Brian Murphy, Leila Wehbe, and Alona Fyshe. Decoding language from the brain. 2018. [10.1](#)
- [305] Masatoshi Nagano, Tomoaki Nakamura, Takayuki Nagai, Daichi Mochihashi, Ichiro Kobayashi, and Wataru Takano. Hvgh: Unsupervised segmentation for high-dimensional time series using deep neural compression and statistical generative model. *Frontiers in Robotics and AI*, 6, 2019. [4.4](#), [4.4](#)
- [306] Myura Nagendran et al. Artificial intelligence versus clinicians: systematic review of design, reporting standards, and claims of deep learning studies. *bmj*, 368, 2020. ISBN: 1756-1833 Publisher: British Medical Journal Publishing Group. [11.1](#)
- [307] Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *AAAI*, 2017. [5.1](#)
- [308] Ramesh Nallapati, Bowen Zhou, Cícero Nogueira dos Santos, Caglar Gülcehre, and Bing Xiang. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *CoNLL*, 2016. [3.3.4](#), [5.1](#)
- [309] Medhini Narasimhan, Anna Rohrbach, and Trevor Darrell. Clip-it! language-guided video summarization. *ArXiv*, abs/2107.00650:13988–14000, 2021. [1.1](#), [5.1](#), [5.1](#)
- [310] Medhini G. Narasimhan, Arsha Nagrani, Chen Sun, Michael Rubinstein, Trevor Darrell, Anna Rohrbach, and Cordelia Schmid. Tl;dw? summarizing instructional videos with task relevance & cross-modal saliency. *ArXiv*, abs/2208.06773, 2022. [5.2](#)
- [311] Shashi Narayan, Shay B. Cohen, and Mirella Lapata. Don’t give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *ArXiv*,

- abs/1808.08745, 2018. [5.1](#)
- [312] Shashi Narayan, Shay B Cohen, and Mirella Lapata. Ranking sentences for extractive summarization with reinforcement learning. In *NAACL*, pages 1747–1759, 2018. [5.2](#)
- [313] Grigorios Nasios, Efthymios Dardiotis, and Lambros Messinis. From broca and wernicke to the neuromodulation era: Insights of brain language networks for neurorehabilitation. *Behavioural Neurology*, 2019, 2019. [10.4.6](#)
- [314] Annamalai Natarajan, Yale Chang, Sara Mariani, Asif Rahman, Gregory Boverman, Shruti Gopal Vij, and Jonathan Rubin. A wide and deep transformer neural network for 12-lead ecg classification. *2020 Computing in Cardiology*, pages 1–4, 2020. [9.2](#)
- [315] Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. Facebook fair’s wmt19 news translation task submission. In *Proc. of WMT*, 2020. [6.2](#)
- [316] Van-Quang Nguyen, Masanori Suganuma, and Takayuki Okatani. Grit: Faster and better image captioning transformer using dual visual features. *ArXiv*, abs/2207.09666, 2022. <https://github.com/davidnvq/grit>. [6.4.1](#), [6.4.3](#), [7.3.1](#), [7.4.3](#), [7.1](#), [7.2](#)
- [317] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *ICML*, 2022. <https://github.com/openai/glide-text2im>. [1.1](#), [6.4.1](#), [6.4.3](#)
- [318] Ana Sofia Nicholls. A neural model for text segmentation. 2021. [5.2](#)
- [319] David Noever and Samantha E. Miller Noever. Reading isn’t believing: Adversarial attacks on multi-modal neurons. *ArXiv*, abs/2103.10480, 2021. [6.1](#), [6.2](#)
- [320] Naoki Nonaka and Jun Seita. In-depth benchmarking of deep neural network architectures for ecg diagnosis. In Ken Jung, Serena Yeung, Mark Sendak, Michael Sjoding, and Rajesh Ranganath, editors, *Proceedings of the 6th Machine Learning for Healthcare Conference*, volume 149 of *Proceedings of Machine Learning Research*, pages 414–439. PMLR, 06–07 Aug 2021. [9.2](#)
- [321] Ole Numssen, Danilo Bzdok, and Gesa Hartwigsen. Functional specialization within the inferior parietal lobes across cognitive domains. *eLife*, 10, 03 2021. [10.4.6](#)
- [322] OpenAI. Gpt-4 technical report. *ArXiv*, abs/2303.08774, 2023. [6.5](#)
- [323] Pablo Ortega and A. Aldo Faisal. Deep learning multimodal fnirs and eeg signals for bimanual grip force decoding. *Journal of Neural Engineering*, 18, 08 2021. [10.2](#)
- [324] Mayu Otani, Yuta Nakashima, Esa Rahtu, J. Heikkilä, and N. Yokoya. Video summarization using deep semantic features. *ArXiv*, abs/1609.08758, 2016. [2.1](#), [3.2](#), [5.2](#)
- [325] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022. [5.1](#)
- [326] John William Paisley, Aimee K. Zaas, Christopher W. Woods, Geoffrey S. Ginsburg, and Lawrence Carin. A stick-breaking construction of the beta process. In *ICML*, 2010. [4.4](#)
- [327] Jia-Yu Pan and Christos Faloutsos. Videocube: A novel tool for video mining and classifica-

- tion. In *International Conference on Asian Digital Libraries*, 2002. [4.2](#)
- [328] Jia-Yu Pan, Hiroyuki Kitagawa, Christos Faloutsos, and Masafumi Hamamoto. Autosplit: Fast and scalable discovery of hidden variables in stream and multimedia databases. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 2004. [4.2](#)
- [329] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002. [5.5.2](#), [6.4.2](#), [7.4.1](#), [8.5.1](#), [9.5.1](#)
- [330] Simone Parisi, Aravind Rajeswaran, Senthil Purushwalkam, and Abhinav Kumar Gupta. The unsurprising effectiveness of pre-trained vision models for control. In *ICML*, 2022. [7.2](#)
- [331] Cheul Young Park, Narae Cha, Soowon Kang, Auk Kim, Ahsan Habib Khandoker, Leontios Hadjileontiadis, Alice Oh, Yong Jeong, and Uichin Lee. K-emocon, a multimodal sensor dataset for continuous emotion recognition in naturalistic conversations. *Scientific Data*, 7, 09 2020. [10.4.1](#), [10.4.2](#)
- [332] Jungin Park, Jiyoung Lee, Ig-Jae Kim, and Kwanghoon Sohn. Sumgraph: Video summarization via recursive graph modeling. In *ECCV*, pages 647–663. Springer, 2020. [5.1](#)
- [333] Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu. On aliased resizing and surprising subtleties in gan evaluation. In *CVPR*, 2022. [6.4.2](#), [6.4.3](#)
- [334] Viorica Patraucean et al. Perception test : A diagnostic benchmark for multimodal models. 2022. [2.1](#)
- [335] Sayak Paul and Pin-Yu Chen. Vision transformers are robust learners. In *AAAI*, 2022. [6.1](#), [6.2](#), [6.2](#)
- [336] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014. [5.2](#)
- [337] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *NAACL*, 2018. [7.2](#)
- [338] Steffen E. Petersen, Paul M. Matthews, Fabian Bamberg, David A. Bluemke, Jane M. Francis, Matthias G. Friedrich, Paul Leeson, Eike Nagel, Sven Plein, and Frank E. Rademakers. Imaging in population science: cardiovascular magnetic resonance in 100,000 participants of UK Biobank-rationale, challenges and approaches. *Journal of Cardiovascular Magnetic Resonance*, 15(1):1–10, 2013. ISBN: 1532-429X Publisher: BioMed Central. [11.4.4](#)
- [339] Steffen E. Petersen, Paul M. Matthews, Jane M. Francis, Matthew D. Robson, Filip Zemrak, Redha Boubertakh, Alistair A. Young, Sarah Hudson, Peter Weale, and Steve Garratt. UK Biobank’s cardiovascular magnetic resonance protocol. *Journal of cardiovascular magnetic resonance*, 18(1):1–7, 2015. ISBN: 1532-429X Publisher: BioMed Central. [11.4.4](#)
- [340] Gabriel Peyré, Marco Cuturi, and Justin M. Solomon. Gromov-wasserstein averaging of kernel and distance matrices. In *ICML*, 2016. [4.4](#)
- [341] Jim Pitman. Poisson–dirichlet and gem invariant distributions for split-and-merge transformations of an interval partition. *Combinatorics, Probability and Computing*, 11:501 – 514, 2002. [4.4](#)
- [342] Yair Poleg, Chetan Arora, and Shmuel Peleg. Temporal segmentation of egocentric videos. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2537–2544,

2014. [4.2](#), [5.2](#)
- [343] Bernd Porr, Luis Howell, Ioannis Stournaras, and Yoav Nir. Popular ecg r peak detectors written in python. 2022. [9.4](#)
- [344] Amy Pu, Hyung Won Chung, Ankur P. Parikh, Sebastian Gehrmann, and Thibault Sellam. Learning compact metrics for mt. In *Conference on Empirical Methods in Natural Language Processing*, 2021. [8.5.1](#)
- [345] Xavier Puig, Kevin Ra, Marko Boben, Jiaman Li, Tingwu Wang, Sanja Fidler, and Antonio Torralba. Virtualhome: Simulating household activities via programs. *arXiv:1806.07011 [cs]*, pages 8494–8502, 06 2018. [7.4.1](#), [7.4.2](#)
- [346] Xuqiang Qiao, Ling Zheng, Yinong Li, Yuqing Ren, Zhida Zhang, Ziwei Zhang, and Lihong Qiu. Characterization of the driving style by state–action semantic plane based on the bayesian nonparametric approach. *Applied Sciences*, 2021. [4.4](#)
- [347] Jieliu Qiu, Franck Dernoncourt, Trung Bui, Zhaowen Wang, D. Zhao, and Hailin Jin. Liveseg: Unsupervised multimodal temporal segmentation of long livestream videos. *ArXiv*, abs/2210.05840, 2022. [1.2](#), [1.3](#), [1.3](#), [1.1](#), [1.3](#), [4](#), [12.1](#), [12.1](#), [12.1](#), [12.2](#)
- [348] Jieliu Qiu, William Jongwon Han, Winfred Wang, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Christos Faloutsos, Lei Li, and Lijuan Wang. Entity6k: A large open-domain evaluation dataset for real-world entity recognition. 2024. [1.1](#), [12.1](#)
- [349] Jieliu Qiu, William Jongwon Han, Jiacheng Zhu, Mengdi Xu, Michael Rosenberg, Emerson Liu, Douglas Weber, and Ding Zhao. Transfer knowledge from natural language to electrocardiography: Can we detect cardiovascular disease through language models? *ArXiv*, abs/2301.09017, 2023. [1.2](#), [1.3](#), [1.1](#), [12.1](#), [12.1](#)
- [350] Jieliu Qiu, Ge Huang, and Tai Sing Lee. Visual sequence learning in hierarchical prediction networks and primate visual cortex. *Advances in neural information processing systems*, 2019. [1.2](#), [9.2](#)
- [351] Jieliu Qiu, Peide Huang, Makiya Nakashima, Jae-Hyeok Lee, Jiacheng Zhu, W. H. Wilson Tang, Po-Heng Chen, Christopher Nguyen, Byung-Hak Kim, Debbie Kwon, Douglas Weber, Ding Zhao, and David Chen. Multimodal representation learning of cardiovascular magnetic resonance imaging. *ArXiv*, abs/2304.07675, 2023. [1.2](#), [1.3](#), [1.3](#), [1.1](#), [1.3](#), [11](#), [12.1](#), [12.1](#), [12.1](#), [12.2](#)
- [352] Jieliu Qiu, W. Liu, and Bao-Liang Lu. Multi-view emotion recognition using deep canonical correlation analysis. *International Conference on Neural Information Processing*, 2018. [2.2](#), [4.4](#), [9.2](#), [10.3.3](#)
- [353] Jieliu Qiu, Andrea Madotto, Zhaojiang Lin, Paul A Crook, Yifan Ethan Xu, Xin Luna Dong, Christos Faloutsos, Lei Li, Babak Damavandi, and Seungwhan Moon. Snapntell: Enhancing entity-centric visual question answering with retrieval augmented multimodal llm. *arXiv preprint arXiv:2403.04735*, 2024. [1.2](#), [1.3](#), [1.3](#), [1.1](#), [12.1](#), [12.1](#), [12.1](#)
- [354] Jieliu Qiu, Xin-Yi Qiu, and Kai Hu. Emotion recognition based on gramian encoding visualization. *Brain Informatics*, 2018. [9.2](#)
- [355] Jieliu Qiu, Mengdi Xu, William Jongwon Han, Seungwhan Moon, and Ding Zhao. Embodied executable policy learning with language-based scene summarization. *NAACL*, 2024. [1.3](#),

[1.1](#), [1.3](#), [12.1](#), [12.1](#), [12.2](#)

- [356] Jieliu Qiu and Wei-Ye Zhao. Data encoding visualization based cognitive emotion recognition with ac-gan applied for denoising. *2018 IEEE 17th International Conference on Cognitive Informatics & Cognitive Computing (ICCI*CC)*, pages 222–227, 2018. [9.2](#)
- [357] Jieliu Qiu, Jiacheng Zhu, William Han, Aditesh Kumar, Karthik Mittal, Claire Jin, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Bo Li, Ding Zhao, and Lijuan Wang. Mmsum: A dataset for multimodal summarization and thumbnail generation of videos. 2023. [1.2](#), [1.3](#), [1.3](#), [1.1](#), [1.3](#), [1.3](#), [5.6](#), [12.1](#), [12.1](#), [12.1](#), [12.2](#), [12.2](#)
- [358] Jieliu Qiu, Jiacheng Zhu, Shiqi Liu, William Jongwon Han, Jingqi Zhang, Chaojing Duan, Michael Rosenberg, Emerson Liu, Douglas Weber, and Ding Zhao. Converting ecg signals to images for efficient image-text retrieval via encoding. *ArXiv*, abs/2304.06286, 2023. [1.1](#), [1.3](#), [12.1](#), [12.1](#)
- [359] Jieliu Qiu, Jiacheng Zhu, Michael Rosenberg, Emerson Liu, and D. Zhao. Optimal transport based data augmentation for heart disease diagnosis and prediction. *ArXiv*, abs/2202.00567, 2022. [9.2](#)
- [360] Jieliu Qiu, Jiacheng Zhu, Mengdi Xu, Franck Deroncourt, Trung Bui, Zhaowen Wang, Bo Li, Ding Zhao, and Hailin Jin. Mhms: Multimodal hierarchical multimedia summarization. *ArXiv*, abs/2204.03734, 2022. [2.1](#), [2.2](#), [5.2](#), [10.3.3](#)
- [361] Jieliu Qiu, Jiacheng Zhu, Mengdi Xu, Franck Deroncourt, Trung Bui, Zhaowen Wang, Bo Li, Ding Zhao, and Hailin Jin. Semantics-consistent cross-domain summarization via optimal transport alignment. volume abs/2210.04722, 2022. [1.2](#), [1.3](#), [1.1](#), [2.1](#), [4.4](#), [5.2](#), [12.1](#), [12.1](#)
- [362] Jieliu Qiu, Jiacheng Zhu, Mengdi Xu, Peide Huang, Michael Rosenberg, Douglas Weber, Emerson Liu, and Ding Zhao. Cardiac disease diagnosis on imbalanced electrocardiography data through optimal transport augmentation. *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2022. [1.1](#), [1.3](#), [12.1](#), [12.1](#)
- [363] Jieliu Qiu, Yi Zhu, Xingjian Shi, F. Wenzel, Zhiqiang Tang, Ding Zhao, Bo Li, and Mu Li. Benchmarking robustness of multimodal image-text models under distribution shift. 2024. [1.2](#), [1.2](#), [1.3](#), [1.3](#), [1.1](#), [1.3](#), [1.3](#), [6.6](#), [12.1](#), [12.1](#), [12.1](#), [12.2](#), [12.2](#)
- [364] Jingyu Quan, Yoshihiro Miyake, and Takayuki Nozawa. Incorporating interpersonal synchronization features for automatic emotion recognition from visual and audio data during communication. *Sensors*, 21:5317, 08 2021. [10.4](#)
- [365] Filip Radenovic, Abhimanyu Dubey, Abhishek Kadian, Todor Mihaylov, Simon Vandenhende, Yash Patel, Yi Wen, Vignesh Ramanathan, and Dhruv Mahajan. Filtering, distillation, and hard negatives for vision-language pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6967–6977, 2023. [8.5.1](#)
- [366] Filip Radenovic, Abhimanyu Dubey, Abhishek Kadian, Todor Mihaylov, Simon Vandenhende, Yash J. Patel, Yi Wen, Vignesh Ramanathan, and Dhruv Kumar Mahajan. Filtering, distillation, and hard negatives for vision-language pre-training. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6967–6977, 2023. [8.5.1](#)

- [367] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. <https://github.com/openai/CLIP>. 2.1, 2.1, 2.2, 6.1, 6.2, 6.4.1, 6.4.3, 8.4.2, 8.5.1, 9.1, 11.1
- [368] Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018. 11.3.1
- [369] Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *ArXiv*, abs/1910.10683, 2019. 3.4, 5.4.2, 5.5, 5.5.3, 7.2
- [370] Sushravya Raghunath, John M. Pfeifer, Alvaro E. Ulloa-Cerna, Arun Nemani, Tanner Carbonati, Linyuan Jing, David P. vanMaanen, Dustin N. Hartzel, Jeffery A. Ruhl, Braxton F. Lagerman, Daniel B. Rocha, Nathan J. Stoudt, Gargi Schneider, Kipp W. Johnson, Noah Zimmerman, Joseph B. Leader, H. Lester Kirchner, Christoph J. Griessenauer, Ashraf Hafez, Christopher W. Good, Brandon K. Fornwalt, and Christopher M. Haggerty. Deep neural networks can predict new-onset atrial fibrillation from the 12-lead ecg and help identify those at risk of atrial fibrillation–related stroke. *Circulation*, 143:1287 – 1298, 2021. 9.2
- [371] Anil Rahate, Rahee Walambe, Sheela Ramanna, and Ketan Kotecha. Multimodal co-learning: challenges, applications with datasets, recent advances and future directions. *Information Fusion*, 81:203–239, 2022. 5.1
- [372] Amir Masoud Rahmani, Efat Yousefpoor, Mohammad Sadegh Yousefpoor, Zahid Mehmood, Amir Haider, Mehdi Hosseinzadeh, and Rizwan Ali Naqvi. Machine learning (ml) in medicine: Review, applications, and challenges. *Mathematics*, 2021. (document), 1.5
- [373] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *ArXiv*, abs/2204.06125, 2022. 6.1, 6.2, 6.4.3
- [374] Marc’Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. *CoRR*, abs/1511.06732, 2015. 7.3.4, 7.3.4
- [375] Anyi Rao, Linning Xu, Yu Xiong, Guodong Xu, Qingqiu Huang, Bolei Zhou, and Dahua Lin. A local-to-global approach to multi-modal movie scene segmentation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10143–10152, 2020. 3.3.1, 3.5, 4.2, 4.5, 4.6, 5.2, 5.4.2, 5.5.2, 5.5.3
- [376] Mirco Ravanelli and Yoshua Bengio. Speaker recognition from raw waveform with sincnet. *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 1021–1028, 2018. 9.2, 9.4
- [377] Vipula Rawte, A. Sheth, and Amitava Das. A survey of hallucination in large foundation models. *ArXiv*, abs/2309.05922, 2023. 8.5.4
- [378] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *ICML*, 2019. 6.2
- [379] Aniketh Janardhan Reddy and Leila Wehbe. Can fmri reveal the representation of syntactic structure in the brain? *bioRxiv*, 2021. 10.1
- [380] Ivegen Redko, Titouan Vayer, Rémi Flamary, and Nicolas Courty. Co-optimal transport.

- ArXiv*, abs/2002.03731, 2020. 4.4
- [381] Amna Rehman and Yasir Al Khalili. Neuroanatomy, occipital lobe, 07 2019. 10.4.6
- [382] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. 4.4, 6.3.2
- [383] Allen Z. Ren, Bharat Govil, Tsung-Yen Yang, Karthik Narasimhan, and Anirudha Majumdar. Leveraging language for accelerated learning of tool manipulation. *ArXiv*, abs/2206.13074, 2022. 7.2, 9.1
- [384] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015. 3.3.5, 4.5
- [385] Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. *CVPR*, pages 1179–1195, 2016. 7.3.4
- [386] Martin Riedl and Chris Biemann. Topictiling: A text segmentation algorithm based on lda. In *ACL 2012*, 2012. 5.2
- [387] Alexander Rives, Siddharth Goyal, Joshua Meier, Demi Guo, Myle Ott, C. Lawrence Zitnick, Jerry Ma, and Rob Fergus. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences of the United States of America*, 118, 2021. 9.1
- [388] Mrigank Rochan, Linwei Ye, and Yang Wang. Video summarization using fully convolutional sequence networks. In *Proceedings of the ECCV (ECCV)*, pages 347–363, 2018. 5.1
- [389] Robin Rombach, A. Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *CVPR*, 2022. <https://github.com/CompVis/stable-diffusion>. 1.1, 1.1, 2.1, 6.1, 6.4.1, 6.4.3
- [390] Daniel Rotman, Dror Porat, and Gal Ashour. Robust and efficient video scene detection using optimal sequential grouping. <https://github.com/ivi-ru/video-scene-detection>, 2016. 4.2, 4.5
- [391] Daniel Rotman, Dror Porat, Gal Ashour, and Udi Barzelay. Optimally grouped deep features using normalized cost for video scene detection. <https://github.com/ivi-ru/video-scene-detection>, 2018. 4.2, 4.5, 4.6
- [392] Sebastian Ruder. Recent advances in language model fine-tuning. 2021. 7.2
- [393] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning internal representations by error propagation. 1986. 9.3
- [394] David Ruppert. The elements of statistical learning: Data mining, inference, and prediction. *Journal of the American Statistical Association*, 99:567 – 567, 2004. 10.4.3
- [395] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115:211–252, 2015. 4.4, 5.4.3
- [396] Barbara Rychalska, Dominika Basaj, Alicja Gosiewska, and P. Biecek. Models in the wild:

- On corruption robustness of neural nlp systems. In *ICONIP*, 2019. [6.1](#), [6.2](#)
- [397] Enas S. Yousif, Azmi Shawkat Abdulbaqi, Abduladheem Zaily Hameed, and Saif Al-din M. N. Electroencephalogram signals classification based on feature normalization. *IOP Conference Series: Materials Science and Engineering*, 928:032028, 11 2020. [10.4.3](#)
- [398] Maham Saeidi, Waldemar Karwowski, Farzad Vasheghani Farahani, Krzysztof Fiok, Redha Taiar, Peter A. Hancock, and Awad Al-Juaid. Neural decoding of eeg signals with machine learning: A systematic review. *Brain Sciences*, 11, 2021. [10.2](#)
- [399] Shagan Sah, Sourabh Kulhare, Allison Gray, Subhashini Venugopalan, Emily Tucker Prud'hommeaux, and Raymond W. Ptucha. Semantic text summarization of long videos. *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 989–997, 2017. [3.1](#)
- [400] Chitwan Saharia et al. Photorealistic text-to-image diffusion models with deep language understanding. *ArXiv*, abs/2205.11487, 2022. [6.5](#)
- [401] Ramon Sanabria, Ozan Caglayan, Shruti Palaskar, Desmond Elliott, Loïc Barrault, Lucia Specia, and Florian Metze. How2: A large-scale dataset for multimodal language understanding. *ArXiv*, abs/1811.00347, 2018. [5.1](#)
- [402] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019. [11.3.1](#)
- [403] M. Saquib Sarfraz et al. Temporally-weighted hierarchical clustering for unsupervised action segmentation. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11220–11229, 2021. [4.2](#), [5.2](#)
- [404] Madeline Chantry Schiappa, Yogesh Singh Rawat, Shruti Vyas, Vibhav Vineet, and Hamid Palangi. Multi-modal robustness analysis against language and visual perturbations. *ArXiv*, abs/2207.02159, 2022. [6.2](#), [6.3.2](#)
- [405] Robert Schmidt. Generative text style transfer for improved language sophistication. 2020. [6.3.2](#)
- [406] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. Laion-5b: An open large-scale dataset for training next generation image-text models. *ArXiv*, abs/2210.08402, 2022. [2.1](#), [8.4.2](#), [8.5.1](#)
- [407] Christoph Schuhmann et al. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *ArXiv*, abs/2111.02114, 2021. [2.1](#)
- [408] Jeanette Schulz-Menger, David A. Bluemke, Jens Bremerich, Scott D. Flamm, Mark A. Fogel, Matthias G. Friedrich, Raymond J. Kim, Florian von Knobelsdorff-Brenkenhoff, Christopher M. Kramer, Dudley J. Pennell, Sven Plein, and Eike Nagel. Standardized image interpretation and post-processing in cardiovascular magnetic resonance - 2020 update. *Journal of Cardiovascular Magnetic Resonance*, 22(1):19, March 2020. [11.1](#)
- [409] Dan Schwartz, Mariya Toneva, and Leila Wehbe. Inducing brain-relevant bias in natural language processing models. In *NeurIPS*, 2019. [10.1](#), [10.2](#)
- [410] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh

- Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *European Conference on Computer Vision*, 2022. [8.2](#), [8.3](#), [8.3.5](#)
- [411] Abigail See, Peter J Liu, and Christopher D Manning. Get to the point: Summarization with pointer-generator networks. In *ACL*, pages 1073–1083, 2017. [3.3.4](#), [3.4](#), [3.5](#), [5.2](#), [5.5.2](#)
- [412] Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. Bleurt: Learning robust metrics for text generation. In *Annual Meeting of the Association for Computational Linguistics*, 2020. [8.5.1](#)
- [413] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128:336–359, 2017. [6.4.3](#)
- [414] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Gallagher, and Tina Eliassi-Rad. Collective classification in network data. *AI Mag.*, 29:93–106, 2008. [5.1](#)
- [415] Jay Sethuraman. A constructive definition of dirichlet priors. 1991. [4.4](#)
- [416] Nur Muhammad (Mahi) Shafiullah, Chris Paxton, Lerrel Pinto, Soumith Chintala, and Arthur D. Szlam. Clip-fields: Weakly supervised semantic fields for robotic memory. *ArXiv*, abs/2210.05663, 2022. [7.2](#)
- [417] Dhruv Shah, Blazej Osinski, Brian Ichter, and Sergey Levine. Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action. *ArXiv*, abs/2207.04429, 2022. [7.2](#), [9.1](#)
- [418] Divya Shanmugam, Davis W. Blalock, Jen J. Gong, and John V. Guttag. Multiple instance learning for ecg risk stratification. *ArXiv*, abs/1812.00475, 2019. [9.2](#)
- [419] Xiaoyu Shen, Yang Zhao, Hui Su, and Dietrich Klakow. Improving latent alignment in text summarization by generalizing the pointer generator. In *EMNLP*, 2019. [3.4](#)
- [420] Sam Shleifer and Alexander M. Rush. Pre-trained summarization distillation. *ArXiv*, abs/2010.13002, 2020. [5.4.2](#), [5.5](#), [5.5.3](#)
- [421] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Cliport: What and where pathways for robotic manipulation. *ArXiv*, abs/2109.12098, 2021. [7.2](#)
- [422] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Perceiver-actor: A multi-task transformer for robotic manipulation. *ArXiv*, abs/2209.05451, 2022. [7.2](#), [9.1](#)
- [423] Panagiotis Sidiropoulos, Vasileios Mezaris, Yiannis Kompatsiaris, Hugo Meinedo, Miguel M. F. Bugalho, and Isabel Trancoso. Temporal video segmentation to scenes using high-level audiovisual features. *IEEE Transactions on Circuits and Systems for Video Technology*, 21:1163–1177, 2011. [4.2](#), [5.2](#)
- [424] Hrituraj Singh, Anshul Nasery, Denil Mehta, Aishwarya Agarwal, Jatin Lamba, and Balaji Vasan Srinivasan. Mimoqa: Multimodal input multimodal output question answering. In *North American Chapter of the Association for Computational Linguistics*, 2021. [8.2](#), [8.3](#), [8.3.5](#)
- [425] Rahul Singh et al. Robustness tests of nlp machine learning models: Search and semantically replace. *ArXiv*, abs/2104.09978, 2021. [6.1](#), [6.2](#), [6.2](#)
- [426] Sandra Śmigiel, Krzysztof Pałczyński, and Damian Ledziński. Ecg signal classification using deep learning techniques based on the ptb-xl dataset. *Entropy*, 23(9):1121, 2021. [9.2](#)

- [427] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP*, pages 1631–1642, 2013. [10.4.1](#), [10.4.2](#)
- [428] Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Ndedi Monekosso, and Paolo Remagnino. Superframes, a temporal video segmentation. *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 566–571, 2018. [4.2](#), [5.2](#)
- [429] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mass: Masked sequence to sequence pre-training for language generation. *arXiv:1905.02450 [cs]*, 06 2019. [7.4.2](#)
- [430] Yale Song, Miriam Redi, Jordi Vallmitjana, and Alejandro Jaimes. To click or not to click: Automatic selection of beautiful thumbnails from videos. *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 2016. [4.5](#), [4.6](#)
- [431] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. Tvsum: Summarizing web videos using titles. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5179–5187, 2015. [3.5](#), [4.1](#), [4.2](#), [4.3](#), [4.3](#), [4.5](#), [5.2](#), [5.2](#), [5.1](#), [5.2](#), [5.3.3](#)
- [432] Yonghao Song, Xueyu Jia, Lie Yang, and Longhan Xie. Transformer-based spatial-temporal feature learning for eeg decoding. *ArXiv*, abs/2106.11170, 2021. [9.2](#)
- [433] Tom’avs Souvcek and Jakub Lokovc. Transnet v2: An effective deep network architecture for fast shot transition detection. volume abs/2008.04838, 2020. [4.5](#)
- [434] Krishna Srinivasan, Karthik Raman, Jiecao Chen, Michael Bendersky, and Marc Najork. Wit: Wikipedia-based image text dataset for multimodal multilingual machine learning. *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021. [8.4.1](#)
- [435] Krishna Srinivasan, Karthik Raman, Anupam Samanta, Ling-Yen Liao, Luca Bertelli, and Michael Bendersky. Quill: Query intent with large language models using retrieval augmentation and multi-stage distillation. In *Conference on Empirical Methods in Natural Language Processing*, 2022. [8.1](#), [8.2](#)
- [436] Matteo Stefanini et al. From show to tell: A survey on deep learning-based image captioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45:539–559, 2021. [7.3.1](#), [7.3.4](#)
- [437] Ethan H. Steinberg, Kenneth Jung, Jason A. Fries, Conor K. Corbin, Stephen R. Pfohl, and Nigam Haresh Shah. Language models are an effective representation learning technique for electronic health record data. *Journal of biomedical informatics*, page 103637, 2021. [9.2](#)
- [438] Brigitte Stemmer and John Connolly. The eeg/erp technologies in linguistic research, 12 2012. [10.1](#)
- [439] Nils Strodthoff, Patrick Wagner, Tobias Schaeffter, and Wojciech Samek. Deep learning for ecg analysis: Benchmarks and insights from ptb-xl. *IEEE Journal of Biomedical and Health Informatics*, 25:1519–1528, 2021. [9.2](#)
- [440] Nataliya Strokina, Wenyan Yang, Joni Pajarinen, Nikolay Serbenyuk, Joni-Kristian Kämäräinen, and Reza Ghabcheloo. Visual rewards from observation for sequential tasks: Autonomous pile loading. *Frontiers in Robotics and AI*, 9, 2022. [7.2](#)
- [441] Alane Suhr, Mike Lewis, James Yeh, and Yoav Artzi. A corpus of natural language for visual

- reasoning. In *ACL*, 2017. [6.4.1](#)
- [442] Heung-II Suk, Seong-Whan Lee, and Dinggang Shen. Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis. *NeuroImage*, 101:569–582, 2014. [11.6](#)
- [443] Kai Sun, Y. Xu, Hanwen Zha, Yue Liu, and Xinhsuai Dong. Head-to-tail: How knowledgeable are large language models (llm)? a.k.a. will llms replace knowledge graphs? *ArXiv*, abs/2308.10168, 2023. [8.5.1](#)
- [444] Sunfox. What does normal ecg and abnormal ecg mean? ([document](#)), [9.1](#)
- [445] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *NIPS*, 2014. [2.1](#), [3.2](#), [5.2](#)
- [446] Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In S. Solla, T. Leen, and K. Müller, editors, *NeurIPS*, volume 12. MIT Press, 1999. [7.3.4](#)
- [447] Alon Talmor, Ori Yoran, Amnon Catav, Dan Lahav, Yizhong Wang, Akari Asai, Gabriel Ilharco, Hannaneh Hajishirzi, and Jonathan Berant. Multimodalqa: Complex question answering over text, tables and images. *ArXiv*, abs/2104.06039, 2021. [8.2](#), [8.3](#), [8.3.5](#)
- [448] Allison C. Tam et al. Semantic exploration from language abstractions and pretrained representations. *ArXiv*, abs/2204.05080, 2022. [7.2](#), [9.1](#)
- [449] Hao Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *ArXiv*, abs/1908.07490, 2019. [7.3.1](#)
- [450] Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. Abstractive document summarization with a graph-based attentional neural model. In *ACL*, pages 1171–1181, 2017. [5.2](#)
- [451] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *ArXiv*, abs/1905.11946, 2019. [5.4.3](#)
- [452] Peggy Tang, Kun Hu, Lei Zhang, Jiebo Luo, and Zhiyong Wang. Tldw: Extreme multimodal summarisation of news videos. *ArXiv*, abs/2210.08481, 2022. [2.1](#), [5.2](#)
- [453] Shitao Tang, Litong Feng, Zhanghui Kuang, Yimin Chen, and Wayne Zhang. Fast video shot transition localization with deep structured models. In *ACCV*, 2018. [4.2](#), [4.5](#), [4.6](#)
- [454] Chongyang Tao, Wei Wu, Can Xu, Wenpeng Hu, Dongyan Zhao, and Rui Yan. Multi-representation fusion network for multi-turn response selection in retrieval-based chatbots. *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 2019. [3.5](#)
- [455] Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. Measuring robustness to natural distribution shifts in image classification. *ArXiv*, abs/2007.00644, 2020. [6.3](#), [6.4.2](#), [6.4.3](#), [6.5](#)
- [456] Yee Whye Teh, Gatsby, and Michael I. Jordan. Hierarchical bayesian nonparametric models with applications. 2008. [4.4](#)
- [457] Yee Whye Teh, Michael I. Jordan, Matthew J. Beal, and David M. Blei. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101:1566 – 1581, 2006. [4.4](#), [4.4](#)
- [458] **J. Qiu**, Jiacheng Zhu, Michael Rosenberg, Emerson Liu, and Ding Zhao. Optimal trans-

- port based data augmentation for heart disease diagnosis and prediction. *arXiv preprint arXiv:2202.00567*, 2022. [1.3](#)
- [459] Mariya Toneva, Tom. Mitchell, and Leila Wehbe. Combining computational controls with natural text reveals new aspects of meaning composition. *bioRxiv*, 2020. [10.1](#)
- [460] Mariya Toneva and Leila Wehbe. Interpreting and improving natural-language processing (in machines) with natural language-processing (in the brain). *ArXiv*, abs/1905.11833, 2019. [10.2](#)
- [461] Atousa Torabi, Niket Tandon, and Leonid Sigal. Learning language-visual embedding for movie understanding with natural-language. *ArXiv*, abs/1609.08124, 2016. [2.1](#)
- [462] Hugo Touvron et al. Llama 2: Open foundation and fine-tuned chat models. *ArXiv*, abs/2307.09288, 2023. [8.5.1](#)
- [463] Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, S. M. Ali Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. In *Neural Information Processing Systems*, 2021. [2.1](#)
- [464] Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962v2*, 2019. [5.4.2](#), [5.5](#), [5.5.3](#)
- [465] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *ArXiv*, abs/1807.03748, 2018. [2.1](#)
- [466] Laurens van der Maaten and Geoffrey E. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008. [4.3](#)
- [467] Laurens JP Van Der Maaten. Visualizing high-dimensional data using t-sne. *J Mach Learn Res*, 9:2579, 2008. [11.5.4](#)
- [468] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *ArXiv*, abs/1706.03762, 2017. [2.2](#), [5.4.3](#), [7.1](#), [7.3.1](#), [9.1](#), [9.3](#), [9.3](#), [10.1](#), [10.3.1](#), [10.4.3](#), [10.4.3](#)
- [469] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. *CVPR*, pages 4566–4575, 2015. [5.5.2](#), [6.4.2](#), [7.4.1](#)
- [470] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio’, and Yoshua Bengio. Graph attention networks. *ArXiv*, abs/1710.10903, 2018. [2.1](#)
- [471] Cédric Villani. Topics in optimal transportation. 2003. [3.2](#)
- [472] Patrick Wagner, Nils Strodthoff, R. Bousseljot, D. Kreiseler, F. Lunze, W. Samek, and T. Schaeffter. Ptb-xl, a large publicly available electrocardiography dataset. *Scientific Data*, 7, 2020. [9.4](#)
- [473] Marilyn A. Walker, Ricky Grant, Jennifer Sawyer, Grace I. Lin, Noah Wardrip-Fruin, and Michael Buell. Perceived or not perceived: Film character models for expressive nlg. In *ICIDS*, 2011. [4.4](#), [4.3](#)
- [474] Marilyn A. Walker, Grace I. Lin, and Jennifer Sawyer. An annotated corpus of film dialogue for learning and characterizing character style. In *LREC*, 2012. [4.4](#), [4.3](#)
- [475] Boxin Wang, Chejian Xu, Shuhang Wang, Zhe Gan, Yu Cheng, Jianfeng Gao, Ahmed Has-

- san Awadallah, and Bo Li. Adversarial glue: A multi-task benchmark for robustness evaluation of language models. *ArXiv*, abs/2111.02840, 2021. [6.1](#), [6.2](#), [6.2](#)
- [476] Jianfeng Wang, Xiaowei Hu, Zhe Gan, Zhengyuan Yang, Xiyang Dai, Zicheng Liu, Yumao Lu, and Lijuan Wang. Ufo: A unified transformer for vision-language representation learning. *ArXiv*, abs/2111.10023, 2021. [4.4](#)
- [477] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool. Temporal segment networks for action recognition in videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(11):2740–2755, 2019. [11.3.2](#)
- [478] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks for action recognition in videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41:2740–2755, 2019. [4.2](#), [4.5](#), [5.2](#)
- [479] Peng Wang, Qi Wu, Chunhua Shen, Anthony R. Dick, and Anton van den Hengel. Fvqa: Fact-based visual question answering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:2413–2427, 2016. [8.2](#)
- [480] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *ICML*, 2022. [6.1](#), [6.4.3](#), [7.3.1](#), [7.4.3](#), [7.1](#), [7.2](#)
- [481] Qinxin Wang, Haochen Tan, Sheng Shen, Michael W. Mahoney, and Zhewei Yao. An effective framework for weakly-supervised phrase grounding. *ArXiv*, abs/2010.05379, 2020. [2.1](#)
- [482] Tianlu Wang, Xuezhi Wang, Yao Qin, Ben Packer, Kang Li, Jilin Chen, Alex Beutel, and Ed H. Chi. Cat-gen: Improving robustness in nlp models via controlled adversarial text generation. In *EMNLP*, 2020. [6.1](#), [6.2](#), [6.2](#)
- [483] Weihang Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, Jiazheng Xu, Bin Xu, Juanzi Li, Yuxiao Dong, Ming Ding, and Jie Tang. Cogvlm: Visual expert for pretrained language models. *ArXiv*, abs/2311.03079, 2023. [8.5](#)
- [484] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *ArXiv*, abs/2002.10957, 2020. [4.4](#)
- [485] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M. Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. pages 2097–2106, 2017. [11.1](#)
- [486] Xuezhi Wang, Haohan Wang, and Diyi Yang. Measure and improve robustness in nlp models: A survey. *ArXiv*, abs/2112.08313, 2022. [6.1](#), [6.2](#), [6.2](#)
- [487] Yizhong Wang, Sujian Li, and Jingfeng Yang. Toward fast and accurate neural discourse segmentation. In *EMNLP*, 2018. [5.2](#)
- [488] Zhenhailong Wang and Heng Ji. Open vocabulary electroencephalography-to-text decoding and zero-shot sentiment classification. *ArXiv*, abs/2112.02690, 2021. [10.1](#), [10.2](#), [10.2](#), [10.3.1](#)

- [489] Zhenzhi Wang, Ziteng Gao, Limin Wang, Zhifeng Li, and Gangshan Wu. Boundary-aware cascade networks for temporal action segmentation. In *ECCV*, 2020. [4.2](#), [5.2](#)
- [490] Zifeng Wang, Zhenbang Wu, Dinesh Agarwal, and Jimeng Sun. Medclip: Contrastive learning from unpaired medical images and text. *arXiv preprint arXiv:2210.10163*, 2022. [11.2](#)
- [491] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision. *arXiv:2108.10904 [cs]*, 08 2021. [6.4.3](#), [7.4.3](#)
- [492] Leila Wehbe, Idan Asher Blank, Cory Shain, Richard Futrell, Roger Philip Levy, Titus von der Malsburg, Nathaniel J. Smith, Edward Gibson, and Evelina Fedorenko. Incremental language comprehension difficulty predicts activity in the language network but not the multiple demand network. *bioRxiv*, 2020. [10.1](#)
- [493] Leila Wehbe, Ashish Vaswani, Kevin Knight, and Tom Mitchell. Aligning context-based statistical models of language with brain activity during reading, 10 2014. [10.2](#)
- [494] Huawei Wei, Bingbing Ni, Yichao Yan, Huanyu Yu, Xiaokang Yang, and Chen Yao. Video summarization via semantic attended networks. In *AAAI*, 2018. [2.1](#), [3.2](#), [5.2](#)
- [495] Jason Wei and Kai Zou. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. In *EMNLP*, 2019. [6.3.2](#)
- [496] Kuba Weimann and Tim O. F. Conrad. Transfer learning for ecg classification. *Scientific Reports*, 11, 2021. [9.2](#)
- [497] F. Wenzel et al. Assaying out-of-distribution generalization in transfer learning. *ArXiv*, abs/2207.09239, 2022. [6.1](#), [6.2](#), [6.5](#)
- [498] Adina Williams, Nikita Nangia, and Samuel R. Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL*, 2018. [6.5](#)
- [499] Jennifer Williams and Leila Wehbe. Behavior measures are predicted by how information is encoded in an individual’s brain. *ArXiv*, abs/2112.06048, 2021. [10.1](#)
- [500] Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256, 1992. [7.3.4](#)
- [501] Rhydian Windsor, Amir Jamaludin, Timor Kadir, and Andrew Zisserman. Vision-language modelling for radiological imaging and reports in the low data regime. *arXiv preprint arXiv:2303.17644*, 2023. [11.2](#)
- [502] Michael Wray, Diane Larlus, Gabriela Csurka, and Dima Damen. Fine-grained action retrieval through multiple parts-of-speech embeddings. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 450–459, 2019. [2.1](#)
- [503] Hao Wu, Jiayuan Mao, Yufeng Zhang, Yuning Jiang, Lei Li, Weiwei Sun, and Wei-Ying Ma. Unified visual-semantic embeddings: Bridging vision and language with structured meaning representations. *CVPR*, pages 6602–6611, 2019. [2.1](#), [6.4.2](#)
- [504] Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. Next-gpt: Any-to-any multimodal llm. *ArXiv*, abs/2309.05519, 2023. [2.1](#)
- [505] Yuxiang Wu and Baotian Hu. Learning to extract coherent summary via deep reinforcement

- learning. In *AAAI*, pages 5602–5609, 2018. [5.2](#)
- [506] Shuwen Xiao, Zhou Zhao, Zijian Zhang, Xiaohui Yan, and Min Yang. Convolutional hierarchical attention network for query-focused video summarization. *arXiv preprint arXiv:2002.03740*, 2020. [5.2](#), [5.2](#)
- [507] Chen Xie, Lucas McCullum, Alistair Johnson, Tom Pollard, Brian Gow, and Benjamin Moody. Waveform database software package (wfdb) for python (version 4.0.0). *PhysioNet*, 2022. [9.4](#)
- [508] Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan Loddon Yuille, and Quoc V. Le. Adversarial examples improve image recognition. *CVPR*, 2019. [6.4.3](#)
- [509] Ning Xie, Farley Lai, Derek Doran, and Asim Kadav. Visual entailment task for visually-grounded language learning. *arXiv preprint arXiv:1811.10582*, 2018. [6.4.1](#)
- [510] Ning Xie, Farley Lai, Derek Doran, and Asim Kadav. Visual entailment: A novel task for fine-grained image understanding. *arXiv preprint arXiv:1901.06706*, 2019. [6.4.1](#)
- [511] Yuexiang Xie, Fei Sun, Yang Deng, Yaliang Li, and Bolin Ding. Factual consistency evaluation for text summarization via counterfactual estimation. In *Conference on Empirical Methods in Natural Language Processing*, 2021. [3.5](#)
- [512] Wayne Xiong, L. Wu, Fil Allea, Jasha Droppo, Xuedong Huang, and Andreas Stolcke. The microsoft 2017 conversational speech recognition system. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5934–5938, 2018. [4.3](#)
- [513] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. *arXiv preprint arXiv:2109.14084*, 2021. [11.2](#)
- [514] Ke Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015. [2.1](#)
- [515] Mengdi Xu, Yuchen Lu, Yikang Shen, Shun Zhang, Ding Zhao, and Chuang Gan. Hyperdecision transformer for efficient online policy adaptation, 2023. [7.2](#)
- [516] Mengdi Xu, Yikang Shen, Shun Zhang, Yuchen Lu, Ding Zhao, Joshua Tenenbaum, and Chuang Gan. Prompting decision transformer for few-shot policy generalization. In *International Conference on Machine Learning*, pages 24631–24645. PMLR, 2022. [7.2](#)
- [517] Ran Xu, Caiming Xiong, Wei Chen, and Jason J. Corso. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In *AAAI*, 2015. [4.4](#)
- [518] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mt5: A massively multilingual pre-trained text-to-text transformer. In *North American Chapter of the Association for Computational Linguistics*, 2020. [5.4.3](#)
- [519] Bin Yan and Mingtao Pei. Clinical-bert: Vision-language pre-training for radiograph diagnosis and reports generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2982–2990, 2022. [11.2](#)

- [520] Fei Yan and Krystian Mikolajczyk. Deep correlation for matching images and text. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3441–3450, 2015. [4.4](#)
- [521] Genshen Yan, Shen Liang, Yanchun Zhang, and Fan Liu. Fusing transformer model with temporal features for ecg heartbeat classification. *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 898–905, 2019. [9.2](#)
- [522] Jianwei Yang, Yonatan Bisk, and Jianfeng Gao. Taco: Token-aware cascade contrastive learning for video-text alignment. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11542–11552, 2021. [2.1](#)
- [523] Jinyu Yang, Jiali Duan, S. Tran, Yi Xu, Sampath Chanda, Liqun Chen, Belinda Zeng, Trishul M. Chilimbi, and Junzhou Huang. Vision-language pre-training with triple contrastive learning. *ArXiv*, abs/2202.10401, 2022. <https://github.com/uta-smile/TCL>. [6.1](#), [6.4.1](#), [6.4.3](#)
- [524] Kangning Yang, Benjamin Tag, Yue Gu, Chaofan Wang, Tilman Dingler, Greg Wadley, and Jorge Goncalves. Mobile emotion recognition via multiple physiological signals using convolution-augmented transformer. *Proceedings of the 2022 International Conference on Multimedia Retrieval*, 06 2022. [10.4](#)
- [525] Nan Yang, Tao Ge, Liang Wang, Binxing Jiao, Daxin Jiang, Linjun Yang, Rangan Majumder, and Furu Wei. Inference with reference: Lossless acceleration of large language models. *ArXiv*, abs/2304.04487, 2023. [8.1](#), [8.2](#)
- [526] Xi Yang et al. Gatortron: A large clinical language model to unlock patient information from unstructured electronic health records. *ArXiv*, abs/2203.03540, 2022. [9.1](#), [9.2](#)
- [527] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *NeurIPS*, 2019. [10.1](#)
- [528] Zhuolin Yang, Wei Ping, Zihan Liu, Vijay Anand Korthikanti, Weili Nie, De-An Huang, Linxi (Jim) Fan, Zhiding Yu, Shiyi Lan, Bo Li, Mingyan Liu, Yuke Zhu, Mohammad Shoeybi, Bryan Catanzaro, Chaowei Xiao, and Anima Anandkumar. Re-vilm: Retrieval-augmented visual language model for zero and few-shot image captioning. *ArXiv*, abs/2302.04858, 2023. [8.1](#), [8.2](#)
- [529] Ziyi Yang, Yuwei Fang, Chenguang Zhu, Reid Pryzant, Dongdong Chen, Yu Shi, Yichong Xu, Yao Qian, Mei Gao, Yi-Ling Chen, Liyang Lu, Yujia Xie, Robert Gmyr, Noel Codella, Naoyuki Kanda, Bin Xiao, Lu Yuan, Takuya Yoshioka, Michael Zeng, and Xuedong Huang. i-code: An integrative and composable multimodal learning framework. *arXiv:2205.01818*, 05 2022. [10.1](#)
- [530] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Exploring visual relationship for image captioning. In *ECCV*, 2018. [2.1](#)
- [531] Michihiro Yasunaga, Armen Aghajanyan, Weijia Shi, Rich James, Jure Leskovec, Percy Liang, Mike Lewis, Luke Zettlemoyer, and Wen tau Yih. Retrieval-augmented multimodal language modeling. *ArXiv*, abs/2211.12561, 2023. [8.2](#)
- [532] Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang,

- Fei Huang, and Jingren Zhou. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration, 2023. [8.5.1](#), [8.5](#), [8.5.4](#)
- [533] Dong Yin, Raphael Gontijo Lopes, Jonathon Shlens, Ekin Dogus Cubuk, and Justin Gilmer. A fourier perspective on model robustness in computer vision. In *NeurIPS*, 2019. [6.1](#), [6.2](#), [6.2](#)
- [534] Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. A survey on multimodal large language models. *ArXiv*, abs/2306.13549, 2023. [2.1](#)
- [535] Peter Young, Alice Lai, Micah Hodosh, and J. Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. [6.4.1](#)
- [536] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *ArXiv*, abs/2205.01917, 2022. [6.1](#), [6.4.3](#)
- [537] Tiezheng Yu, Wenliang Dai, Zihan Liu, and Pascale Fung. Vision guided generative pre-trained language models for multimodal abstractive summarization. *ArXiv*, abs/2109.02401, 2021. [5.4.3](#), [5.4.3](#)
- [538] Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim. End-to-end concept word detection for video captioning, retrieval, and question answering. *CVPR*, pages 3261–3269, 2017. [2.1](#)
- [539] Lu Yuan et al. Florence: A new foundation model for computer vision. *ArXiv*, abs/2111.11432, 2021. [2.1](#)
- [540] S. Yuan, K. Bai, Liqun Chen, Yizhe Zhang, Chenyang Tao, C. Li, Guoyin Wang, R. Henao, and L. Carin. Weakly supervised cross-domain alignment with optimal transport. *BMVC*, 2020. [2.2](#), [3.2](#), [3.3.5](#), [3.3.5](#), [4.4](#), [4.4](#), [10.3.3](#)
- [541] Yitian Yuan, Tao Mei, Peng Cui, and Wenwu Zhu. Video summarization by learning deep side semantic embedding. *IEEE Transactions on Circuits and Systems for Video Technology*, 29:226–237, 2019. [2.1](#), [3.2](#), [5.2](#)
- [542] Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. Tensor fusion network for multimodal sentiment analysis. In *EMNLP*, pages 1103–1114, 2017. [3.4](#), [5.5.3](#)
- [543] Rowan Zellers, Ari Holtzman, Matthew E. Peters, Roozbeh Mottaghi, Aniruddha Kembhavi, Ali Farhadi, and Yejin Choi. Piglet: Language grounding through neuro-symbolic interaction in a 3d world. In *ACL*, 2021. [7.2](#), [9.1](#)
- [544] Andy Zeng, Adrian S. Wong, Stefan Welker, Krzysztof Choromanski, Federico Tombari, Aavek Purohit, Michael S. Ryoo, Vikas Sindhwani, Johnny Lee, Vincent Vanhoucke, and Peter R. Florence. Socratic models: Composing zero-shot multimodal reasoning with language. *ArXiv*, abs/2204.00598, 2022. [7.2](#), [7.2](#), [9.1](#)
- [545] Andrew Zhai and Hao-Yu Wu. Classification is a strong baseline for deep metric learning. In *BMVC*, 2019. [11.3.2](#)
- [546] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. *CVPR*, 2022. [2.1](#)

- [547] Bowen Zhang, Hexiang Hu, and Fei Sha. Cross-modal and hierarchical modeling of video and text. In *ECCV*, 2018. [2.1](#)
- [548] Haoxin Zhang, Zhimin Li, and Qinglin Lu. Better learning shot boundary detection via multi-task. *Proceedings of the 29th ACM International Conference on Multimedia*, 2021. [4.2](#), [5.2](#)
- [549] HongJiang Zhang, Atreyi Kankanhalli, and Stephen W. Smoliar. Automatic partitioning of full-motion video. *Multimedia Systems*, 1:10–28, 1993. [5.4.2](#), [5.5.3](#)
- [550] Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. *ArXiv*, abs/1912.08777, 2019. [3.4](#), [5.4.2](#), [5.5](#), [5.5.3](#)
- [551] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman. Video summarization with long short-term memory. In *ECCV*, 2016. [4.5](#), [5.1](#)
- [552] Litian Zhang, Xiaoming Zhang, Junshu Pan, and Feiran Huang. Hierarchical cross-modality semantic correlation learning model for multimodal summarization. In *AAAI*, 2022. [2.1](#), [3.2](#), [5.2](#)
- [553] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. *CVPR*, 2021. [6.1](#)
- [554] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. BERTscore: Evaluating text generation with bert. *ArXiv*, abs/1904.09675, 2019. [3.5](#), [5.5.2](#), [9.5.1](#)
- [555] Xingxing Zhang, Furu Wei, and Ming Zhou. HIBERT: Document level pre-training of hierarchical bidirectional transformers for document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5059–5069, Florence, Italy, July 2019. Association for Computational Linguistics. [5.2](#), [5.2](#)
- [556] Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D. Manning, and Curtis P. Langlotz. Contrastive learning of medical visual representations from paired images and text. In *Machine Learning for Healthcare Conference*, pages 2–25. PMLR, 2022. [11.2](#), [11.6](#)
- [557] Zhengkun Zhang, Xiaojun Meng, Yasheng Wang, Xin Jiang, Qun Liu, and Zhenglu Yang. Unims: A unified framework for multimodal summarization with knowledge distillation. In *AAAI*, 2021. [2.1](#), [5.2](#)
- [558] Bin Zhao, Haopeng Li, Xiaoqiang Lu, and Xuelong Li. Reconstructive sequence-graph network for video summarization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2793–2801, 2021. [5.1](#)
- [559] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2933–2942, 2017. [4.2](#), [5.2](#)
- [560] Stephan Zheng, Yang Song, Thomas Leung, and Ian J. Goodfellow. Improving the robustness of deep neural networks via stability training. *CVPR*, 2016. [6.1](#), [6.2](#), [6.2](#)
- [561] Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. Extractive summarization as text matching. *arXiv preprint arXiv:2004.08795*, 2020. [5.1](#)

- [562] Binggui Zhou, Guanghua Yang, Zheng Shi, and Shaodan Ma. Natural language processing for smart healthcare. *ArXiv*, abs/2110.15803, 2021. [9.2](#)
- [563] Daquan Zhou, Zhiding Yu, Enze Xie, Chaowei Xiao, Anima Anandkumar, Jiashi Feng, and José Manuel Álvarez. Understanding the robustness in vision transformers. In *ICML*, 2022. [6.1](#), [6.2](#)
- [564] Feng Zhou, Fernando De la Torre, and Jessica K. Hodgins. Hierarchical aligned cluster analysis for temporal clustering of human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35:582–596, 2013. [4.2](#), [5.2](#)
- [565] Jian Zhou, Fei-Yue Wang, and Da jun Zeng. Hierarchical dirichlet processes and their applications: a survey. *Acta Automatica Sinica*, 37:389–407, 2011. [4.4](#)
- [566] Kaiyang Zhou, Yu Qiao, and Tao Xiang. Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward. In *AAAI*, pages 7582–7589, 2018. [3.4](#), [5.1](#), [5.2](#), [5.2](#)
- [567] Kaiyang Zhou, T. Xiang, and A. Cavallaro. Video summarisation by classification with deep reinforcement learning. In *BMVC*, 2018. [5.2](#), [5.2](#)
- [568] Luwei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason J. Corso, and Jianfeng Gao. Unified vision-language pre-training for image captioning and vqa. *ArXiv*, abs/1909.11059, 2019. [7.3.1](#)
- [569] Luwei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, pages 7590–7598, 2018. [5.3.1](#)
- [570] P. Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. Attention-based bidirectional long short-term memory networks for relation classification. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 2016. [10.4.3](#)
- [571] Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu. Multi-turn response selection for chatbots with deep attention matching network. In *ACL*, 2018. [3.5](#)
- [572] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigt-4: Enhancing vision-language understanding with advanced large language models. *ArXiv*, abs/2304.10592, 2023. [8.5](#)
- [573] Jiacheng Zhu, Aritra Guha, Dat Do, Mengdi Xu, XuanLong Nguyen, and Ding Zhao. Functional optimal transport: map estimation and domain adaptation for functional data. *arXiv preprint arXiv:2102.03895*, 2021. [3.2](#)
- [574] Jiacheng Zhu, Jieli Qiu, Aritra Guha, Zhuolin Yang, XuanLong Nguyen, Bo Li, and Ding Zhao. Interpolation for robust learning: Data augmentation on wasserstein geodesics. In *International Conference on Machine Learning*, 2023. [1.1](#), [1.3](#), [12.1](#), [12.1](#)
- [575] Jiacheng Zhu, Jieli Qiu, Zhuolin Yang, Douglas Weber, Michael A. Rosenberg, Emerson Liu, Bo Li, and Ding Zhao. Geocg: Data augmentation via wasserstein geodesic perturbation for robust electrocardiogram prediction. In *MLHC*, 2022. [1.1](#), [1.3](#), [2.2](#), [9.2](#), [9.4](#), [10.3.3](#), [12.1](#), [12.1](#)
- [576] Junnan Zhu, Haoran Li, Tianshan Liu, Yu Zhou, Jiajun Zhang, and Chengqing Zong. Msmo: Multimodal summarization with multimodal output. In *EMNLP*, 2018. [1.1](#), [2.1](#), [3.1](#), [3.2](#), [3.4](#)

[3.5](#), [5.1](#), [5.2](#), [5.1](#), [5.5.3](#)

- [577] Junnan Zhu, Yu Zhou, Jiajun Zhang, Haoran Li, Chengqing Zong, and Changliang Li. Multimodal summarization with guidance of multimodal reference. In *AAAI*, 2020. [3.4](#)
- [578] Wencheng Zhu, Jiwen Lu, Jiahao Li, and Jie Zhou. Dsnet: A flexible detect-to-summarize network for video summarization. *IEEE Transactions on Image Processing*, 30:948–962, 2020. [5.1](#)
- [579] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: deformable transformers for end-to-end object detection. *CoRR*, abs/2010.04159, 2020. [7.4.3](#)
- [580] Mohammadreza Zolfaghari, Yi Zhu, Peter Gehler, and Thomas Brox. Crossclr: Cross-modal contrastive learning for multi-modal video representations. In *ICCV*, 2021. [1.1](#), [2.1](#)
- [581] Sandra Śmigiel, Krzysztof Pałczyński, and Damian Ledziński. Ecg signal classification using deep learning techniques based on the ptb-xl dataset. *Entropy*, 23, 2021. [9.4](#)