Forensic Analysis of Observed Social Networks

Daniele Bellutta

CMU-S3D-26-103
May 2026

Software and Societal Systems Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Kathleen M. Carley, Chair
Hirokazu Shirado
Nynke M. D. Niezink
Gian Maria Campedelli (Fondazione Bruno Kessler)

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Societal Computing.

Copyright © 2026 Daniele Bellutta

This work was supported in part by the US Office of Naval Research through grants N000141612324, N000141712675, N000141812106, N000141812108, N000142112229, N000142112765, and N000142412414, the US Army through grant W911NF20D0002, and Cognizant Technology Solutions through a gift in support of building trust and safety for online platforms. This work was also supported by the John S and James L Knight Foundation and the Center for Informed Democracy and Social-cybersecurity. Additional support was provided by the Center for Computational Analysis of Social and Organizational Systems.

The views and conclusions contained herein are those of the author and should not be interpreted as representing the official policies of the US Office of Naval Research, the US Army, the Knight Foundation, Cognizant Technology Solutions, or the US government.

Keywords: network science, social network analysis, link prediction, uncertainty analysis

Abstract

Analysts studying relationships between people often rely on information that is inadequate or inaccurate. Collecting information to map out a social network may not be immediately possible, can overlook certain connections between individuals, and may even record ties incorrectly. These problems are particularly problematic when studying networks representing the veiled activities of covert groups like troll farms and terrorist organizations, since they intentionally try to hide their behavior. This thesis develops methods for understanding underlying network activity from inadequate or inaccurate information by detecting possible anomalies in online account creations, identifying missing or spurious connections in observed social networks, and measuring the uncertainty in metrics calculated on imperfect network data. Evaluations using real and simulated networks demonstrate that these techniques can provide early warnings of online influence campaigns, emend complex network data that vary over time, and put bounds on the relative importance of covert actors. This thesis then applies these approaches to a sample of social media messages coupled with more complete records of the authors' activities in order to explore how these techniques can improve the reliability of subsequent analysis. Ultimately, this research provides guidance on how network scientists can adjust their analysis to cope with missing or spurious data, such as when social media analysts study concerning online activity. Finally, this thesis highlights how law enforcement and government intelligence agencies can improve their analysis of social networks representing clandestine behavior.

Acknowledgments

This thesis would not have been possible without the guidance and support of my doctoral advisor, Dr. Kathleen M. Carley. I am thankful to have had such an understanding advisor who has accommodated significant family obligations and given me freedom to explore my own interests. I am also grateful to Dr. Hirokazu Shirado, Dr. Nynke Niezink, and Dr. Gian Maria Campedelli for serving on my thesis committee and providing constructive feedback for improving the quality of this research. This thesis is stronger because of their contributions.

This work would also not have been possible without my master's advisor, Dr. V. S. Subrahmanian, who helped me develop my research skills through attentive mentorship. I also thank the many individuals who have guided my research endeavors during my internships, including (but not limited to) Dr. Jeffrey Plaut, Dr. Yonggyu Gim, and Dr. John Hunt. Several of my past teachers and professors also deserve credit for getting me to this point, including Dr. Paula Baker, Dr. Daniel Verdier, Dr. Yinqian Zhang, Dr. Benjamin McKean, Dr. Peter Winkler, Dr. Andrew Campbell, Dr. Gregory Neat, and many more.

Many other people have assisted my research along the way, including Dr. Joshua Uyheng, Dr. Janice Blane, Dr. Catherine King, Dr. David Beskow, Sienna Watkins, Michael Kowalchuck, Jeffrey Reminga, Dawn Robertson, Anja Subasic, Dr. Chiara Pulice, and Dr. Youdinghuan (David) Chen. I am fortunate to have written articles with many of them. In addition, Dr. Matthew Hicks deserves my many thanks for having been so collaborative and charitable during our time as fellow teaching assistants.

For our many stimulating conversations, I am grateful to all the past and present students of our research group, particularly Peter, Iain, and Evan. Several visiting students have also been good friends to me during recent years, namely Melania, Veronica, and Ariadna. Our university staff have also been critical to my work, including Tom, Cole, Connie, Alisha, Dabney, Aaron (Buck), Richelle, Bernadette, and many other people.

My time as a PhD student would not have been nearly as enjoyable without my wonderful office mates: Dr. Courtney Miller, Dr. Janice Blane, Dr. Rebecca Marigliano, and Chanel. Courtney is one of the most respectful people I know and has constantly been a motivating exemplar of diligence. Janice was a beacon of positivity and warmth, always brightening the office mood. And Reba is the kind of person who, despite the challenges in her own life, unfailingly asks how a person is feeling and truly cares to hear an honest response. Not to be outdone, Chanel almost single-pawedly sustained my well-being on many difficult days, and I hope the attention I paid her made up for never having given her any treats. My office mates have made coming to campus a source of happiness, and I will deeply miss spending my days with them.

I am also greatly indebted to the other people who have been friends to me during my time at CMU. Tori has been an excellent model for placing more trust in the world and persevering with less frustration. Samsara has repeatedly demonstrated

how to generously support friends and be more open with others. Samantha has made Herculean efforts to include me in social occasions, and I am very grateful for her solidarity. Josh was critical in helping me pass our statistics class and (more importantly) was always amiable and kind. Meryl has been a worthy opponent in our gift exchange and occasional tête-à-tête, regularly providing comedic relief. Lynnette's caring postcards have raised my spirits during many challenging periods, often coming at fortuitous times. Anisha's stream of compliments and fascinating conversation served as refreshing breaks at the tail end of my studies. And Bianca's fearless self-expression inspired me to pursue my own creative interests and taught me more about myself. I have learned so much from each of them and dearly hope to hear more from them all in the future.

I would not be where I am now without my friends from Ohio State. Harry has been an unwavering wellspring of encouragement, thoughtfulness, and respite, often in moments so serendipitous as to be almost spooky. Alec has proven an admirable template for caring for living things, and he has consistently supplied genuine faith and reassurance. James has regularly offered constructive support, and I am thankful for his compassion and understanding. Amjad frequently imparted ethical direction and counsel, and he was the best roommate I have ever had. Henry has exemplified great affability and tolerance over the years since our physics experiments, and Max has provided lasting fellowship and academic guidance ever since our late nights studying calculus. And Eric contributed many interesting conversations in the years following our origins learning political theory. I owe much more to my friends than can be expressed in words, and I hope these bonds will endure for many dozens more years.

I am of course grateful to my parents, Paolo and Paola, and to my brother Marco for having supported me through three decades of life. I also thank my other family members, including my uncle Claudio, my aunts Luisa and Chiara, and my cousin Silvia, for having been positive influences on my childhood and adulthood. And I would not be the person I am today without the excellent examples set by my grandmother Carmela and grandfather Bruno. Finally, I thank all the children of Seio who were my friends as a kid and helped make my summers the best parts of my childhood.

I am thankful to these and many other people, so I apologize to anyone I have neglected to appropriately credit in this list. The world has shown me great kindness throughout my life, too much to fit even in a doctoral thesis.

Contents

1 Observed Social Networks & Forensics **1**
- 1.1 Introduction . . . 1
- 1.2 Objectives . . . 2
- 1.3 Definitions . . . 2
- 1.4 Scope . . . 2
- 1.5 Background . . . 3
  - 1.5.1 Imperfect Network Data . . . 3
  - 1.5.2 Covert Network Analysis . . . 5
  - 1.5.3 Social Media Intelligence . . . 6
- 1.6 Reconsidering Social Monitoring . . . 7
- 1.7 Data . . . 9
  - 1.7.1 Real Networks . . . 9
  - 1.7.2 Stylized Networks . . . 11
- 1.8 Outline . . . 11

2 Forewarning Impending Coordination **13**
- 2.1 Introduction . . . 13
  - 2.1.1 Motivation . . . 13
  - 2.1.2 Objectives . . . 14
- 2.2 Methods . . . 15
  - 2.2.1 Data Collection . . . 15
  - 2.2.2 Network Construction . . . 15
  - 2.2.3 Burst Detection . . . 15
  - 2.2.4 Regression Analysis . . . 16
  - 2.2.5 Measured Features . . . 17
- 2.3 Results . . . 20
  - 2.3.1 Communication with Others . . . 20
  - 2.3.2 Talk Among Contemporaries . . . 21
  - 2.3.3 Hashtag Usage Patterns . . . 21
  - 2.3.4 Agreement on Controversial Issues . . . 22
  - 2.3.5 Synchronous Behavior . . . 22
  - 2.3.6 Bot Activity . . . 23
  - 2.3.7 Sharing of Low-Credibility Sources . . . 23
- 2.4 Discussion . . . 23

2.5 Conclusion . . . 25
2.5.1 Limitations . . . 25
2.5.2 Contributions . . . 25

3 Emending Multidimensional Networks **27**
3.1 Introduction . . . 27
3.1.1 Motivation . . . 27
3.1.2 Objectives . . . 28
3.2 Methods . . . 28
3.2.1 Tensor Perturbation Method . . . 28
3.2.2 Benchmark Algorithms . . . 30
3.2.3 Performance Measurement . . . 30
3.2.4 Inferring Simulated Missing Links . . . 31
3.2.5 Inferring Missing Links in Covert Networks . . . 32
3.2.6 Identifying Simulated Spurious Links . . . 36
3.3 Results . . . 37
3.3.1 Simulated Missing Link Inference . . . 37
3.3.2 Covert Missing Link Inference . . . 39
3.3.3 Simulated Spurious Link Identification . . . 43
3.4 Discussion . . . 48
3.4.1 Missing Links . . . 48
3.4.2 Spurious Links . . . 49
3.5 Conclusion . . . 50
3.5.1 Limitations . . . 50
3.5.2 Contributions . . . 51

4 Bounding Network Metrics **53**
4.1 Introduction . . . 53
4.1.1 Motivation . . . 53
4.1.2 Objectives . . . 54
4.2 Methods . . . 54
4.2.1 Generating Simulated Networks . . . 54
4.2.2 Bootstrapping Networks . . . 55
4.2.3 Constructing Confidence Intervals . . . 56
4.2.4 Evaluating Interval Quality . . . 56
4.3 Results . . . 57
4.3.1 Average Degree Centrality . . . 57
4.3.2 Average Betweenness Centrality . . . 59
4.3.3 Triangle Density . . . 59
4.3.4 Degree Centralities . . . 60
4.3.5 Betweenness Centralities . . . 63
4.3.6 Runtime . . . 63
4.4 Discussion . . . 63
4.5 Conclusion . . . 66

4.5.1 Limitations . . . 66
4.5.2 Contributions . . . 66

5 Improving Analytical Robustness **69**
5.1 Introduction . . . 69
5.1.1 Motivation . . . 69
5.1.2 Objectives . . . 69
5.2 Improving the Accuracy of Node Centralities . . . 70
5.2.1 Background . . . 70
5.2.2 Methods . . . 70
5.2.3 Results . . . 71
5.2.4 Discussion . . . 71
5.3 Stabilizing Node Centralities Over Time . . . 73
5.3.1 Background . . . 73
5.3.2 Methods . . . 73
5.3.3 Results . . . 74
5.3.4 Discussion . . . 74
5.4 Identifying Subjects for Data Collection . . . 76
5.4.1 Background . . . 76
5.4.2 Methods . . . 76
5.4.3 Results . . . 78
5.4.4 Discussion . . . 79
5.5 Reducing Uncertainty in Network Data . . . 79
5.5.1 Background . . . 79
5.5.2 Methods . . . 79
5.5.3 Results . . . 80
5.5.4 Discussion . . . 88
5.6 Conclusion . . . 88
5.6.1 Limitations . . . 88
5.6.2 Contributions . . . 89

6 Implications for Network Analysts **91**
6.1 Contributions . . . 91
6.1.1 Empirical . . . 91
6.1.2 Methodological . . . 92
6.2 Significance for Practitioners . . . 92
6.2.1 Network Scientists . . . 92
6.2.2 Law Enforcement . . . 93
6.2.3 Social Media Analysts . . . 95
6.3 Limitations . . . 95
6.4 Future Directions . . . 96

A Data Collection & Processing for 2020 US Elections **99**
A.1 Terms Tracked in Data Collection . . . 99
A.2 Hashtags Used to Initialize Stance Detection . . . 100
A.3 Low-Credibility Web Domains . . . 101

B Regression Results for Account Creation Bursts in 2020 US Election Tweets **107**
B.1 Communication With Others . . . 107
B.2 Talk Amongst Contemporaries . . . 107
B.3 Hashtag Use . . . 107
B.4 Agreement on Political Issues . . . 107
B.5 Hashtag Synchrony . . . 108
B.6 Bot Activity . . . 108
B.7 Sharing of Low-Credibility Sources . . . 108

C Link Inference Methodology **137**
C.1 Data Processing . . . 137
C.1.1 Timeline Collection for French Terror Tweets . . . 137
C.1.2 Topic Construction for Islamic Network Forum Data . . . 138
C.2 Link Inference . . . 138
C.2.1 Tensor Perturbation Method . . . 138
C.2.2 Parameter Optimization . . . 138

D Simulated Missing Link Inference Results **141**
D.1 Multiplex Stylized Networks . . . 141
D.1.1 Small-World Networks . . . 141
D.1.2 Scale-Free Networks . . . 141
D.1.3 Uniformly Random Networks . . . 141
D.2 Dynamic Multiplex Stylized Networks . . . 141
D.2.1 Small-World Networks . . . 141
D.2.2 Scale-Free Networks . . . 141
D.2.3 Uniformly Random Networks . . . 142

E Covert Missing Link Inference Results **167**
E.1 Multiplex Covert Networks . . . 167
E.1.1 Belo Horizonte Co-Offenses . . . 167
E.1.2 Operazione Montagna Surveillance . . . 167
E.1.3 Multiplex French Terror Network . . . 167
E.2 Dynamic Multiplex Covert Networks . . . 167
E.2.1 Dynamic Multiplex French Terror Network . . . 167
E.2.2 ViNSAR Terrorist Rivalries . . . 167
E.2.3 Islamic Network Online Forum . . . 168

F Simulated Spurious Link Identification Results **181**
F.1 Multiplex Stylized Networks . . . 181
F.1.1 Small-World Networks . . . 181
F.1.2 Scale-Free Networks . . . 181
F.1.3 Uniformly Random Networks . . . 181
F.2 Dynamic Multiplex Stylized Networks . . . 181
F.2.1 Small-World Networks . . . 181
F.2.2 Scale-Free Networks . . . 181
F.2.3 Uniformly Random Networks . . . 182

G Confidence Interval Results **207**
G.1 Average Degree Centrality . . . 207
G.2 Average Betweenness Centrality . . . 207
G.3 Triangle Density . . . 207
G.4 Degree Centralities . . . 207
G.5 Betweenness Centralities . . . 207
G.6 Runtime . . . 208

H Regression Results for Account Creation Bursts in 2020 French Terror Tweets **227**
H.1 Bot Activity . . . 227
H.2 Sharing of Low-Credibility Sources . . . 227

Bibliography **235**

List of Figures

1.1 Precision and Recall for Predicting New Discussion Participants in Islamist Forum 8

2.1 Daily Account Creations Extracted from 2020 US Election Tweets 17
2.2 Partisan Hashtag Usage in 2020 US Election Tweets 19
2.3 Example Communication Networks Among Contemporary Accounts 22

3.1 Missing Link Inference on 100-Node Multiplex Stylized Networks 38
3.2 Missing Link Inference on 100-Node Dynamic Multiplex Stylized Networks . . . 39
3.3 Link Inference on the Belo Horizonte Network 40
3.4 Link Inference on the Operazione Montagna Network 41
3.5 Link Inference on the Multiplex French Terror Network 42
3.6 Link Inference on the Dynamic Multiplex French Terror Network 42
3.7 Link Inference on the ViNSAR Terrorist Rivalries Network 44
3.8 Link Inference on the Islamic Network Forum 45
3.9 Spurious Link Identification on 100-Node Multiplex Stylized Networks 46
3.10 Spurious Link Identification on 100-Node Dynamic Multiplex Stylized Networks 47
3.11 Degree-Weighted Precision on the ViNSAR Terrorist Rivalries Network 50

4.1 Confidence Interval Coverage for Average Degree Centrality 58
4.2 Confidence Interval Coverage for Average Betweenness Centrality 60
4.3 Confidence Interval Coverage for Triangle Density 61
4.4 Confidence Interval Coverage for Node-Level Degree Centralities 62
4.5 Confidence Interval Coverage for Node-Level Betweenness Centralities 64
4.6 Runtime for Bootstrapping Stylized Networks 65

5.1 Accuracy of Centrality Measures Before and After Link Inference 72
5.2 Daily Account Creations Extracted from 2020 French Terror Tweets 77
5.3 Change in Degree Interval Width When Collecting by Decreasing Degree 82
5.4 Change in Betweenness Interval Width When Collecting by Decreasing Degree . 82
5.5 Change in Degree Interval Width When Collecting by Increasing Degree 83
5.6 Change in Betweenness Interval Width When Collecting by Increasing Degree . . 83
5.7 Change in Degree Interval Width When Collecting by Decreasing Degree Interval Width 84
5.8 Change in Betweenness Interval Width When Collecting by Decreasing Degree Interval Width 84

5.9 Change in Degree Interval Width When Collecting by Decreasing Betweenness . 85
5.10 Change in Betweenness Interval Width When Collecting by Decreasing Betweenness 85
5.11 Change in Degree Interval Width When Collecting by Increasing Betweenness . . 86
5.12 Change in Betweenness Interval Width When Collecting by Increasing Degree . . 86
5.13 Change in Degree Interval Width When Collecting by Decreasing Betweenness Interval Width 87
5.14 Change in Betweenness Interval Width When Collecting by Decreasing Betweenness Interval Width 87

6.1 Precision and Recall for Predicting New Discussion Participants with Link Prediction 94

C.1 Anti-France Hashtags 137

D.1 AUPRC for Inferring Any Missing Links in Multiplex Small-World Networks . . 142
D.2 Precision for Inferring Any Missing Links in Multiplex Small-World Networks . 143
D.3 AUROC for Inferring Any Missing Links in Multiplex Small-World Networks . . 144
D.4 Runtime for Inferring Any Missing Links in Multiplex Small-World Networks . . 145
D.5 AUPRC for Inferring Any Missing Links in Multiplex Scale-Free Networks . . . 146
D.6 Precision for Inferring Any Missing Links in Multiplex Scale-Free Networks . . 147
D.7 AUROC for Inferring Any Missing Links in Multiplex Scale-Free Networks . . . 148
D.8 Runtime for Inferring Any Missing Links in Multiplex Scale-Free Networks . . . 149
D.9 AUPRC for Inferring Any Missing Links in Multiplex Uniformly Random Networks 150
D.10 Precision for Inferring Any Missing Links in Multiplex Uniformly Random Networks 151
D.11 AUROC for Inferring Any Missing Links in Multiplex Uniformly Random Networks 152
D.12 Runtime for Inferring Any Missing Links in Multiplex Uniformly Random Networks 153
D.13 AUPRC for Inferring Any Missing Links in Dynamic Multiplex Small-World Networks 154
D.14 Precision for Inferring Any Missing Links in Dynamic Multiplex Small-World Networks 155
D.15 AUROC for Inferring Any Missing Links in Dynamic Multiplex Small-World Networks 156
D.16 Runtime for Inferring Any Missing Links in Dynamic Multiplex Small-World Networks 157
D.17 AUPRC for Inferring Any Missing Links in Dynamic Multiplex Scale-Free Networks 158
D.18 Precision for Inferring Any Missing Links in Dynamic Multiplex Scale-Free Networks 159

D.19 AUROC for Inferring Any Missing Links in Dynamic Multiplex Scale-Free Networks . . . 160
D.20 Runtime for Inferring Any Missing Links in Dynamic Multiplex Scale-Free Networks . . . 161
D.21 AUPRC for Inferring Any Missing Links in Dynamic Multiplex Uniformly Random Networks . . . 162
D.22 Precision for Inferring Any Missing Links in Dynamic Multiplex Uniformly Random Networks . . . 163
D.23 AUROC for Inferring Any Missing Links in Dynamic Multiplex Uniformly Random Networks . . . 164
D.24 Runtime for Inferring Any Missing Links in Dynamic Multiplex Uniformly Random Networks . . . 165

E.1 AUPRC for Inferring Missing Links in the Belo Horizonte Network . . . 169
E.2 AUROC and Runtime for Inferring Missing Links in the Belo Horizonte Network 169
E.3 AUPRC for Inferring Missing Links in the Operazione Montagna Network . . . 170
E.4 Precision for Inferring Missing Links in the Operazione Montagna Network . . . 171
E.5 AUROC and Runtime for Inferring Missing Links in the Operazione Montagna Network . . . 172
E.6 AUPRC for Inferring Missing Links in the Multiplex French Terror Network . . . 172
E.7 Precision for Inferring Missing Links in the Multiplex French Terror Network . . . 173
E.8 AUROC and Runtime for Inferring Missing Links in the Multiplex French Terror Network . . . 173
E.9 AUPRC for Inferring Missing Links in the Dynamic Multiplex French Terror Network . . . 174
E.10 Precision for Inferring Missing Links in the Dynamic Multiplex French Terror Network . . . 174
E.11 AUROC and Runtime for Inferring Missing Links in the Dynamic Multiplex French Terror Network . . . 175
E.12 AUPRC for Inferring Missing Links in the ViNSAR Network . . . 175
E.13 Precision for Inferring Missing Links in the ViNSAR Network . . . 176
E.14 AUROC and Runtime for Inferring Missing Links in the ViNSAR Network . . . 176
E.15 AUPRC for Inferring Missing Links on the Islamic Network Forum . . . 177
E.16 Precision for Inferring Missing Links on the Islamic Network Forum . . . 178
E.17 AUROC and Runtime for Inferring Missing Links on the Islamic Network Forum 179

F.1 AUPRC for Identifying Any Spurious Links in Multiplex Small-World Networks 182
F.2 Precision for Identifying Any Spurious Links in Multiplex Small-World Networks 183
F.3 AUROC for Identifying Any Spurious Links in Multiplex Small-World Networks 184
F.4 Runtime for Identifying Any Spurious Links in Multiplex Small-World Networks 185
F.5 AUPRC for Identifying Any Spurious Links in Multiplex Scale-Free Networks . 186
F.6 Precision for Identifying Any Spurious Links in Multiplex Scale-Free Networks . 187
F.7 AUROC for Identifying Any Spurious Links in Multiplex Scale-Free Networks . 188
F.8 Runtime for Identifying Any Spurious Links in Multiplex Scale-Free Networks . 189

F.9 AUPRC for Identifying Any Spurious Links in Multiplex Uniformly Random Networks . . . 190
F.10 Precision for Identifying Any Spurious Links in Multiplex Uniformly Random Networks . . . 191
F.11 AUROC for Identifying Any Spurious Links in Multiplex Uniformly Random Networks . . . 192
F.12 Runtime for Identifying Any Spurious Links in Multiplex Uniformly Random Networks . . . 193
F.13 AUPRC for Identifying Any Spurious Links in Dynamic Multiplex Small-World Networks . . . 194
F.14 Precision for Identifying Any Spurious Links in Dynamic Multiplex Small-World Networks . . . 195
F.15 AUROC for Identifying Any Spurious Links in Dynamic Multiplex Small-World Networks . . . 196
F.16 Runtime for Identifying Any Spurious Links in Dynamic Multiplex Small-World Networks . . . 197
F.17 AUPRC for Identifying Any Spurious Links in Dynamic Multiplex Scale-Free Networks . . . 198
F.18 Precision for Identifying Any Spurious Links in Dynamic Multiplex Scale-Free Networks . . . 199
F.19 AUROC for Identifying Any Spurious Links in Dynamic Multiplex Scale-Free Networks . . . 200
F.20 Runtime for Identifying Any Spurious Links in Dynamic Multiplex Scale-Free Networks . . . 201
F.21 AUPRC for Identifying Any Spurious Links in Dynamic Multiplex Uniformly Random Networks . . . 202
F.22 Precision for Identifying Any Spurious Links in Dynamic Multiplex Uniformly Random Networks . . . 203
F.23 AUROC for Identifying Any Spurious Links in Dynamic Multiplex Uniformly Random Networks . . . 204
F.24 Runtime for Identifying Any Spurious Links in Dynamic Multiplex Uniformly Random Networks . . . 205

G.1 Confidence Interval Coverage for Average Degree Centrality on Small-World Networks . . . 208
G.2 Confidence Interval Coverage for Average Degree Centrality on Scale-Free Networks . . . 209
G.3 Confidence Interval Coverage for Average Degree Centrality on Uniformly Random Networks . . . 210
G.4 Confidence Interval Coverage for Average Betweenness Centrality on Small-World Networks . . . 211
G.5 Confidence Interval Coverage for Average Betweenness Centrality on Scale-Free Networks . . . 212

G.6 Confidence Interval Coverage for Average Betweenness Centrality on Uniformly Random Networks . . . 213
G.7 Confidence Interval Coverage for Triangle Density on Small-World Networks . . 214
G.8 Confidence Interval Coverage for Triangle Density on Scale-Free Networks . . . 215
G.9 Confidence Interval Coverage for Triangle Density on Uniformly Random Networks . . . 216
G.10 Confidence Interval Coverage for Node-Level Degree Centralities on Small-World Networks . . . 217
G.11 Confidence Interval Coverage for Node-Level Degree Centralities on Scale-Free Networks . . . 218
G.12 Confidence Interval Coverage for Node-Level Degree Centralities on Uniformly Random Networks . . . 219
G.13 Confidence Interval Coverage for Node-Level Betweenness Centralities on Small-World Networks . . . 220
G.14 Confidence Interval Coverage for Node-Level Betweenness Centralities on Scale-Free Networks . . . 221
G.15 Confidence Interval Coverage for Node-Level Betweenness Centralities on Uniformly Random Networks . . . 222
G.16 Runtimes for Bootstrapping Small-World Networks . . . 223
G.17 Runtimes for Bootstrapping Scale-Free Networks . . . 224
G.18 Runtimes for Bootstrapping Uniformly Random Networks . . . 225

List of Tables

1.1 Data Index 9

2.1 Average Marginal Effects for Account Creation Bursts in 2020 US Elections . . . 21

3.1 Multiplex Covert Networks for Link Inference 33
3.2 Dynamic Multiplex Covert Networks for Link Inference 34

5.1 Cumulative Networks Formed from French Terror Tweets 73
5.2 Temporal Stability of Centrality Measures Before and After Link Inference . . . 75
5.3 Average Marginal Effects for Burst Accounts in French Terror Discussions 78

B.1 Coefficients for Mean Neighborhood Cosine Similarity with a PFA of 0.25 109
B.2 Coefficients for Mean Neighborhood Cosine Similarity with a PFA of 0.30 109
B.3 Coefficients for Mean Neighborhood Cosine Similarity with a PFA of 0.35 110
B.4 Effects for Mean Neighborhood Cosine Similarity with a PFA of 0.25 110
B.5 Effects for Mean Neighborhood Cosine Similarity with a PFA of 0.30 111
B.6 Effects for Mean Neighborhood Cosine Similarity with a PFA of 0.35 111
B.7 Coefficients for Mean Clustering Coefficient with a PFA of 0.25 112
B.8 Coefficients for Mean Clustering Coefficient with a PFA of 0.30 112
B.9 Coefficients for Mean Clustering Coefficient with a PFA of 0.35 113
B.10 Effects for Mean Clustering Coefficient with a PFA of 0.25 113
B.11 Effects for Mean Clustering Coefficient with a PFA of 0.30 114
B.12 Effects for Mean Clustering Coefficient with a PFA of 0.35 114
B.13 Coefficients for Communication Density with a PFA of 0.25 115
B.14 Coefficients for Communication Density with a PFA of 0.30 115
B.15 Coefficients for Communication Density with a PFA of 0.35 116
B.16 Effects for Communication Density with a PFA of 0.25 116
B.17 Effects for Communication Density with a PFA of 0.30 117
B.18 Effects for Communication Density with a PFA of 0.35 117
B.19 Coefficients for Mean Hashtag Cosine Similarity with a PFA of 0.25 118
B.20 Coefficients for Mean Hashtag Cosine Similarity with a PFA of 0.30 118
B.21 Coefficients for Mean Hashtag Cosine Similarity with a PFA of 0.35 119
B.22 Effects for Mean Hashtag Cosine Similarity with a PFA of 0.25 119
B.23 Effects for Mean Hashtag Cosine Similarity with a PFA of 0.30 120
B.24 Effects for Mean Hashtag Cosine Similarity with a PFA of 0.35 120

B.25 Coefficients for Standard Deviation of Mail-In Voting Stances with a PFA of 0.25 121
B.26 Coefficients for Standard Deviation of Mail-In Voting Stances with a PFA of 0.30 121
B.27 Coefficients for Standard Deviation of Mail-In Voting Stances with a PFA of 0.35 122
B.28 Effects for Standard Deviation of Mail-In Voting Stances with a PFA of 0.25 . . . 122
B.29 Effects for Standard Deviation of Mail-In Voting Stances with a PFA of 0.30 . . . 123
B.30 Effects for Standard Deviation of Mail-In Voting Stances with a PFA of 0.35 . . . 123
B.31 Coefficients for Standard Deviation of Mask Wearing Stances with a PFA of 0.25 124
B.32 Coefficients for Standard Deviation of Mask Wearing Stances with a PFA of 0.30 124
B.33 Coefficients for Standard Deviation of Mask Wearing Stances with a PFA of 0.35 125
B.34 Effects for Standard Deviation of Mask Wearing Stances with a PFA of 0.25 . . . 125
B.35 Effects for Standard Deviation of Mask Wearing Stances with a PFA of 0.30 . . . 126
B.36 Effects for Standard Deviation of Mask Wearing Stances with a PFA of 0.35 . . . 126
B.37 Coefficients for Proportion of Account Pairs Synchronizing Hashtags with a PFA of 0.25 . . . 127
B.38 Coefficients for Proportion of Account Pairs Synchronizing Hashtags with a PFA of 0.30 . . . 127
B.39 Coefficients for Proportion of Account Pairs Synchronizing Hashtags with a PFA of 0.35 . . . 128
B.40 Effects for Proportion of Account Pairs Synchronizing Hashtags with a PFA of 0.25 . . . 128
B.41 Effects for Proportion of Account Pairs Synchronizing Hashtags with a PFA of 0.30 . . . 129
B.42 Effects for Proportion of Account Pairs Synchronizing Hashtags with a PFA of 0.35 . . . 129
B.43 Coefficients for Proportion of Bot Accounts with a PFA of 0.25 . . . 130
B.44 Coefficients for Proportion of Bot Accounts with a PFA of 0.30 . . . 130
B.45 Coefficients for Proportion of Bot Accounts with a PFA of 0.35 . . . 131
B.46 Effects for Proportion of Bot Accounts with a PFA of 0.25 . . . 131
B.47 Effects for Proportion of Bot Accounts with a PFA of 0.30 . . . 132
B.48 Effects for Proportion of Bot Accounts with a PFA of 0.35 . . . 132
B.49 Coefficients for Proportion of Accounts Sharing Low-Credibility Sources with a PFA of 0.25 . . . 133
B.50 Coefficients for Proportion of Accounts Sharing Low-Credibility Sources with a PFA of 0.30 . . . 133
B.51 Coefficients for Proportion of Accounts Sharing Low-Credibility Sources with a PFA of 0.35 . . . 134
B.52 Effects for Proportion of Accounts Sharing Low-Credibility Sources with a PFA of 0.25 . . . 134
B.53 Effects for Proportion of Accounts Sharing Low-Credibility Sources with a PFA of 0.30 . . . 135
B.54 Effects for Proportion of Accounts Sharing Low-Credibility Sources with a PFA of 0.35 . . . 135

C.1 Link Inference Parameters . . . 140

H.1 Coefficients for Bot probability with a PFA of 0.25 228
H.2 Coefficients for Bot probability with a PFA of 0.30 228
H.3 Coefficients for Bot probability with a PFA of 0.35 229
H.4 Effects for Bot probability with a PFA of 0.25 229
H.5 Effects for Bot probability with a PFA of 0.30 230
H.6 Effects for Bot probability with a PFA of 0.35 230
H.7 Coefficients for Proportion of posts with low-credibility links with a PFA of 0.25 231
H.8 Coefficients for Proportion of posts with low-credibility links with a PFA of 0.30 231
H.9 Coefficients for Proportion of posts with low-credibility links with a PFA of 0.35 232
H.10 Effects for Proportion of posts with low-credibility links with a PFA of 0.25 . . . 232
H.11 Effects for Proportion of posts with low-credibility links with a PFA of 0.30 . . . 233
H.12 Effects for Proportion of posts with low-credibility links with a PFA of 0.35 . . . 233

Chapter 1

Observed Social Networks & Forensics

1.1 Introduction

Though already in use for decades in law enforcement [105, 180] and national security [63] domains, social network analysis has grown evermore significant since the popularization of online social media [97]. Indeed, such platforms have become an invaluable source of information for both intelligence analysts [83] and police officers [183] across the world, who increasingly rely on automated tools for identifying evidence and threats on social media [124, 148, 171]. At the same time, democratic societies have faced greater threats from covert actors attempting to influence their elections [10, 21, 32], with social media companies struggling to curtail the spread of disinformation [21, 55]. These developments have only amplified the importance of studying these clandestine groups, making the analysis of covert networks even more important among government agencies and social media analysts.

At the heart of these applications of social network analysis lies the fundamental problem of potential mismatches between the information known to observers and the activity being studied [156]. Criminal organizations purposefully conceal their pursuits, and social media companies restrict access to their data. Collecting information on human subjects is prone to bias and error, and the scale of big data makes it difficult to know what information actually matters. These issues result in observers either not having access to desired information or collecting data that are incomplete or contain spurious interactions, potentially sabotaging the accuracy of subsequent analysis and conclusions. Indeed, several studies have shown that missing or spurious nodes and edges can severely impact analytical metrics calculated on a network and its nodes [28, 78, 140, 199, 203]. Accurately understanding the stories that lie behind these veiled social networks consequently requires strategies for dealing with situations of severe data scarcity or observation error, an undertaking that requires a forensic mindset [98] and bridges network science with signal processing, machine learning, and statistics.

For some tasks such as filling in missing network connections, prior research has already produced several practical techniques. However, many of these methodologies have not been tested in ways that realistically represent the processes that can cause pieces of covert networks to be missing [214], and only a small minority have been developed for use with rich social media data that are both dynamic (i.e., change over time) and multiplex (i.e., contain multiple types of

relationships). Meanwhile, network scientists have also lacked clear guidance on how to measure uncertainty in their analysis [156], and social media companies have found themselves in constant conflict with malevolent actors who conceal their coordinated behavior within the sea of global online activity [10, 32, 55]. These issues in coping with imperfect information perpetuate potential inaccuracies in covert network analysis, which should be concerning to professionals in policing and intelligence.

1.2 Objectives

This thesis encompasses two principal objectives: (a) to develop methods for understanding network activity from inadequate or inaccurate information and (b) to explore how these techniques might benefit the analysis of covert networks. Pursuant to those goals, the research explores three approaches to coping with inadequate or inaccurate network data: (i) detecting anomalies in available information, (ii) identifying missing or spurious connections, and (iii) quantifying resulting analytical uncertainty. The thesis then explores the use of these techniques on a single collection of potentially extremist social media activity. Consequently, the main chapters of this thesis focus on the following research questions.

§2. Can abnormal spikes in social media account creations warn of impending coordination?
§3. How can missing and spurious ties be discovered in networks that are both dynamic and multiplex?
§4. How can appropriate confidence intervals be formed for metrics calculated on observed networks?
§5. How might techniques for handling imperfect network data improve the robustness of social network analysis?

The remainder of this chapter defines relevant terms, sets the scope of the work, and provides background information on pertinent prior research.

1.3 Definitions

Parts of this thesis discuss networks of higher dimensionality than standard networks. For these purposes, *multiplex* networks are those that contain multiple kinds of connections between nodes, and *dynamic* networks are those that change over time. Finally, the term *multidimensional* refers to the broad category of networks that may be multiplex, dynamic, or both.

Separately, some of this research refers to the task of link inference or link prediction. Though the terms are often used interchangeably in scientific literature, a clear distinction will be made for the purposes of this thesis: *link prediction* will refer to the forecasting of future connections, whereas *link inference* will refer to the task of filling in connections in current or past data.

1.4 Scope

In developing methods for analyzing imperfect networks, this research focuses on social networks. Though much of network science can be applied across network types, restricting this

work to social networks suitably limits the otherwise intractable scope of possible networks. Additionally, this thesis does not consider the modeling of networks using variables specified by an analyst, since formulating such models requires the kind of explicit theorizing and testing that can be challenging for people who are not network scientists or lack the time to evaluate different models. Existing work has also already developed strategies for handling missing data in such situations [93, 99, 107, 113, 116]. Finally, this research considers imperfections in a network's edges, not its nodes. As reflected by a much larger body of prior research, working with unseen or incorrect edges is a simpler (yet still significant) problem than that of hidden or superfluous nodes, not least because the possible state space is well defined for the former but not the latter. Though this restriction limits the applicability of this research in certain situations, it makes the topic more manageable.

1.5 Background

Though forensic science is frequently defined in the context of investigating crime from available evidence [9], the associated mindset has been likened to other scientific efforts [98] and extended to the exercise of discerning intelligence through data analytics [61] and social network analysis [4]. Understanding underlying network activity from inadequate or inaccurate information can be fraught with error, particularly when actors are purposefully trying to remain covert or when analysts wish to glean intelligence from global social media platforms.

1.5.1 Imperfect Network Data

Analytical Impact

Several prior studies have examined the reliability of network analysis performed on samples of a true network. Though Costenbader and Valente [52] found that some network centrality measures are relatively robust to sampling between fifty and eighty percent of the true network, Borgatti et al. [28] observed that the accuracy of centrality measures decreases with additional error in a network (i.e., added or removed edges or nodes), making it more difficult to correctly identify the top-ranking nodes in a network. Overall, they suggested that centrality metrics should be reasonably robust as long as the amount of network error is less than ten percent. Ficara et al. [76] later reached a similar conclusion when removing edges from mafia networks but cautioned that removing even a small proportion of nodes could significantly degrade understanding of a network. Frantz et al. [79] further found that the effects of missing information depend upon the topology of the true network, with certain centralities in scale-free and small-world networks exhibiting severe vulnerability to node removal and node addition errors, respectively. Looking instead at network-level measures, Kossinets [114] used simulations to show that such metrics can be drastically altered by missing information and that the resulting biases are not random. In contrast to Borgatti et al. [28], the simulations conducted by Smith and Moody [177] indicated that different measures suffered from missing nodes at different rates, with closeness centrality and betweenness centrality being more sensitive than degree-based centralities. However, in a different study that randomly removed both nodes and edges from criminal networks, Berlusconi

[22] instead concluded that betweenness centrality was relatively robust to missing information.

The few empirical studies in this area that have not relied exclusively on simulations have born out concerns that missing information can severely impact the results of network analysis. Two studies comparing Twitter's one-percent streaming sample to the platform's more complete (but expensive) "firehose" and "gardenhose" revealed that Twitter's sampled streams did not provide random samples and could not be relied upon to draw accurate conclusions for certain types of analysis [140, 199]. For example, users' in-degree centralities calculated on a four-week streamed sample could only identify seventy-three of the top one hundred users in the complete firehose [140]. In a different area, Goossen and Paruchuri [85] compared collaboration networks formed from patent records and confirmed earlier findings [114] that the differences in network metrics were not random. In the realm of covert networks, De Moor et al. [59] compared co-offender networks formed from police records to corresponding networks augmented with crime-scene DNA evidence and found that betweenness centrality suffered greater error than other centrality measures, confirming a report by Smith and Moody [177] but contrasting again with those from Borgatti et al. [28] and Berlusconi [22]. Though some of these results contradict each other, it overall remains clear that missing more than a small portion of a network can severely impact the conclusions drawn using standard network analysis tools like centrality measures.

In addition to suffering from missing data, an observed network can also contain connections that either do not exist in reality or are irrelevant to the phenomenon being studied. Though these spurious links can appear in observed networks because of well-known issues like false reports in human subjects research [34], they have received notably less research attention than missing edges [167, 217, 220]. Borgatti et al. [28] conducted simulations on random networks and concluded that spurious edges degraded node centrality measures more than missing edges. However, when Wang et al. [203] ran similar simulations on real social networks, they instead found that (except for clustering coefficient) spurious edges did not reduce the accuracy of node metrics as much as missing edges. Other research has focused instead on the relevance of edges rather than their existence, such as a study reporting that the structure of a communication network can change dramatically depending on the minimum communication frequency that an analyst chooses to count as a connection [49]. Removing irrelevant edges from networks has also garnered attention as a way of improving the tractability of algorithms operating on large networks, including those from social media [84, 174].

Types of Missingness

Most research simulating missing network data has only considered information that is referred to as *missing completely at random* (MCAR), like when edges are removed uniformly at random. But the field of survey research has also contemplated other ways in which information can come to be missing in collected data [126, 166, 169]. For example, information can still be *missing at random* (MAR) if its missingness is related to the correctly observed data [126, 166, 169]. For network data, this would mean that an edge's probability of being missing depends on observed edges, known node attributes, or other collected information [6]. Finally, data can be *missing not at random* (MNAR) if its missingness depends on unobserved factors [126, 166, 169], such as if an edge is missing because the data collection process ignored connections related to unknown node characteristics. All these types of missingness are important to study in network contexts.

1.5.2 Covert Network Analysis

History

Despite the surge of interest in analyzing criminal networks since the terrorist attacks of 11 September 2001, the use of network analysis to study covert activity has a much longer history, dating to as early as the US Army's use of traffic analysis to scrutinize signal communications during the Second World War [200]. By the 1970s, early link analysis had spread to police departments, where investigators mapped the relationships between suspected criminals to aid in understanding the operations of organized crime [94, 180]. This continued into the current era, when after 2001 government interest in network analysis was reinvigorated by studies of the network of terrorists responsible for the 9/11 attacks in the United States [101, 117]. Throughout the past few decades, network scientists have also studied several other kinds of illegal or otherwise hidden activity, often known as *dark* or *covert* networks.

Research in Academia

Covert network analysis has grown into a remarkably large field of academic research covering networks of terrorist groups [18, 43, 117], human traffickers [39], wildlife poachers [92], and more. Naturally, understanding the structure and operations of illicit groups has been of significant interest, such as studying how operational security influences the organization of criminal groups [139] and how centrality measures might identify leaders in such organizations [37]. This concern for the operation of criminal organizations has also inspired the statistical modelling of groups' network structures to disentangle the effects of relevant factors [40]. Other researchers have instead created computational models to predict the future actions [184] or evolution [31] of such groups, and previous research has also explored whether sociological theory applies to relational changes between terrorist groups [18]. Even more proactive work has considered how governments might carry out interventions upon criminal organizations, such as selecting individuals to remove from a terrorist network in order to hamper its operations [43, 181].

Before studying a covert network, however, analysts must first obtain the network information itself. Consequently, several studies have tackled the problem of acquiring network data on illicit activity. These efforts have spanned from conducting ethnographic studies [109] and scraping online forums [158] to scouring arrest records [41, 53] and aggregating police wiretaps [45, 76]. Scholars have also sought to highlight how these different collection methods can introduce different biases into the constructed network, such as wiretap data over-representing the connections of surveilled individuals in a kind of "spotlight" effect [41]. This means that connections in police data are often MNAR, since their missingness depends both on the data collection process and on unseen social factors. In order to counteract some of these data collection problems and fill in some of the missing information, covert network researchers have also gotten involved in the field of link prediction [23], which seeks to develop methods for identifying unseen or yet-to-be-formed links between the nodes in a network [118, 128].

Unfortunately, little academic research has studied covert networks that are dynamic [33, 73] or multiplex [77]. This means that most techniques for covert network analysis cannot take advantage of the additional information available in dynamic and multiplex data like social media

networks. Given the difficulty of obtaining data on covert actors, effective methods for analyzing such data must make use of as much of that scarce information as possible.

Adoption in Policing and Intelligence

Despite the decades-long use of link analysis by police investigators and intelligence analysts to map relationships between suspected criminals [94, 180], few of the more recent network analysis techniques appear to have been adopted by law enforcement. Studies involving intelligence analysts in police departments have documented both their use of centrality measures to quantify individuals' relative importance and also their examination of a network's k-cores to identify its core and peripheral members [33, 66]. However, more complex techniques such as those for destabilizing organizations have seemingly not been embraced by these professionals. Instead, Burcher and Whelan [33] found that in addition to understanding the structure of criminal networks, intelligence analysts in Australian law enforcement liked to use network analysis to discover new lines of enquiry for an investigation. More advanced techniques for social network analysis like those for discovering missing connections between suspects could therefore help intelligence analysts in these efforts [23, 38, 53, 72].

A few potential reasons could explain why more recent network science research has not been adopted by the policing and intelligence communities. Research on covert network analysis often reaches across disciplines [210] in ways that can be challenging for professionals with expertise in other areas. Additionally, the sheer number of methods and algorithms that have been developed, particularly for inferring missing links [118] and identifying social media bots [55], can make it difficult to know which techniques to ultimately learn and implement, which is only compounded when there is a lack of consensus on which methods are most appropriate for a given situation [82]. Nonetheless, the most significant hindrance to the adoption of advanced social network analysis in policing has been a lack of training on the requisite theories and tools [33]. In a study conducted at the Dutch Police Academy, Duijn and Klerks [66] found that after significant instruction, intelligence analysts successfully applied social network analysis to identifying additional informants in a cannabis cultivation network. Proper training could therefore allow intelligence analysts in law enforcement to benefit from social network analysis in their work.

1.5.3 Social Media Intelligence

Emergence from Open-Source Intelligence

Often defined as the collection, analysis, and exploitation of publicly available information, the origin of open-source intelligence (OSINT) as a methodical discipline can be traced as far back as the US Civil War [27]. Since that time, the field has been revolutionized by the proliferation of the Internet and subsequent popularization of online social media platforms [83]. Initially reliant on public sources like newspapers, OSINT now takes advantage of the mass of information posted publicly on social media every day. Aside from making more information accessible to analysts, this radical change has also made OSINT available to consumers beyond government intelligence agencies, such as investigative journalists, risk management professionals, law

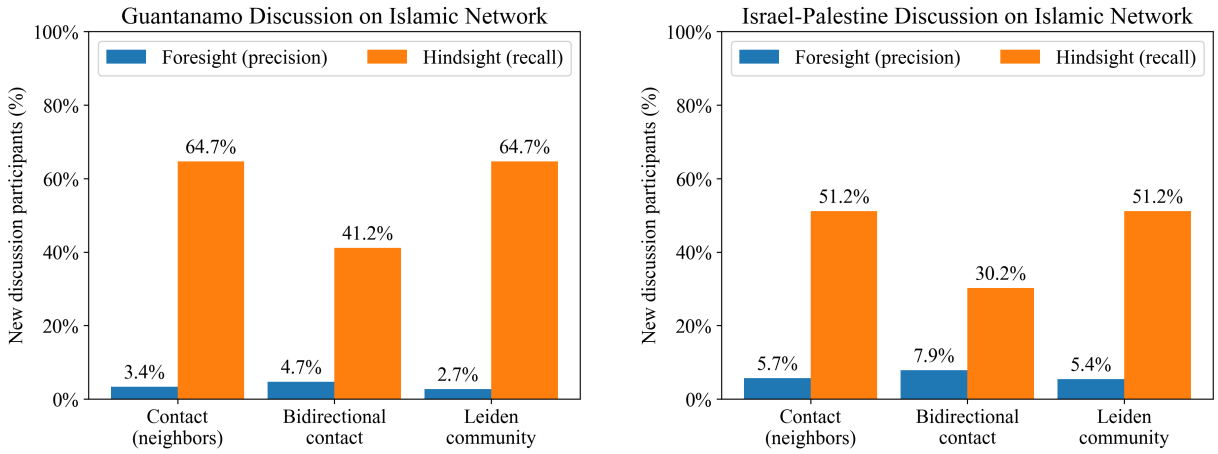
enforcement officers, and (unfortunately) social engineering attackers [35]. The explosion of publicly available information following the rise of online social media platforms has given rise to the field of social media intelligence (SOCMINT or SMI) [138, 149], which some regard as a new discipline of intelligence [149] but others would consider a new form of open-source intelligence [135, 192]. Alas, the mass of information obtainable on social media has come at the cost of greater difficulties in identifying the truly useful information and verifying its accuracy [135]. This means that intelligence analysts must increasingly rely on software tools and computational methods for analyzing social media [124, 148, 171].

Monitoring from Law Enforcement

The explosion of publicly available information following the rise of online social media platforms has given rise to the field of social media intelligence (SOCMINT or SMI) [138, 149], which some regard as a new discipline of intelligence [149] but others would consider a new form of open-source intelligence (OSINT) [135, 192]. The increasing significance of social media intelligence has in turn made social network analysis a foundation of research on online misinformation and extremism. Though this new field of social cybersecurity [42] has garnered significant government interest [142], that does not seem to have stimulated significantly greater use of social network analysis among intelligence analysts studying social media, who largely focus on manually searching for online content [131] or employ artificial intelligence tools for identifying concerning content or criminal evidence [124, 148, 171]. However, social network analysis has made its way into the underlying logic employed by some of these government monitoring tools. In particular, software used by law enforcement to identify users at risk of radicalization has been known to flag users in the communication neighborhoods of known extremists [124]. This strategy appears to be a simplification of academic findings on understanding and predicting radicalization, many of which regard a person's sense of isolation and the attitudes of one's peers as being salient risk factors for radicalization [29, 30, 208]. Perhaps as a result of this simplification, these social media monitoring tools have been criticized for lacking nuance and depth, with some claiming that their simplistic algorithms yield too many false positives [124].

1.6 Reconsidering Social Monitoring

To further demonstrate the need for additional research on social network forensics, basic versions of the strategies known to be used by social media monitoring tools were evaluated using messages posted from 2004 to 2010 to the Islamic Network, a former English-language forum for Muslims [197]. As described in Appendix C, keywords were used to identify forum posts on topics that were more likely to contain extremist discussion, namely Guantanamo Bay and the Israel-Palestine conflict. Communication networks corresponding to each topic as well as to the forum overall were formed by creating ties from a comment's author to the authors of the original post and any preceding comments, an approach also employed by Phillips et al. [158]. A comparison was then performed between the users who discussed a topic during its peak three-month period and those who newly joined that discussion in the three months after the peak, with the goal being to understand whether the latter group had had prior contact with the former group.



(a) Guantanamo Bay

(b) Israel-Palestine Conflict

Figure 1.1: The success rates of various strategies for forecasting which forum users will join a discussion in the three months following its peak discussion period. Foresight (precision) indicates the proportion of users predicted by each strategy who actually ended up joining the discussion. Hindsight (recall) indicates the proportion of users joining the discussion who could have been predicted using each strategy. The contact strategy looks at who had prior contact with existing discussion participants, whereas the Leiden strategy identifies users who were in the Leiden communities [191] of any existing discussion participants.

As shown in Figure 1.1, the "hindsight" results demonstrate that the new people who joined a discussion often had previous contact with existing discussion participants, thereby supporting the approach used by some social media monitoring tools. More than 64% of the users who joined the Guantanamo Bay discussion had previously replied to or received replies from existing discussion participants, a number that drops to 41% when only considering users who had had both such kinds of prior contact. Using the Leiden community detection algorithm [191] yielded similar results, with almost 59% of new Guantanamo Bay discussion participants having been in the same communities as existing participants. The results for discussion of the Israel-Palestine conflict were slightly lower but still remarkably high. This seems to indicate that the logic employed by social media monitoring tools might indeed be able to flag people who may end up joining concerning conversations.

However, the same results shown in Figure 1.1 also reveal that these approaches only appear viable from the point of view of hindsight. If one instead considers the actual prediction performance by counting how many of the potentially flagged users really ended up joining a discussion in the future, the results are much less encouraging. From this perspective of foresight, only 3.4% of those who had prior contact with people discussing Guantanamo Bay during its peak actually joined the discussion in the following three months, which is a far cry from the corresponding hindsight result of nearly 65%. The success rate at correctly flagging new participants increases to 4.7% when requiring bidirectional contact with existing participants, but this is still a remarkably low true positive rate. The results were slightly better for discussion of the Israel-Palestine conflict, with 7.9% of users who had both replied to and received replies from existing partici-

Data	Definition	§1	§2	§3	§4	§5
Islamic Network online forum	§3, §C, [197]	•		•		
2020 US election tweets	§2, [13]		•			
2020 French terror tweets	§3, §C, [25]			•		•
Belo Horizonte co-offenses	§3, [189]			•		
Operazione Montagna surveillance	§3, [45, 76]			•		
ViNSAR terrorist rivalries	§3, [51]			•		
Multidimensional stylized networks	§3, [16]				•	
Stylized networks	§4, [16]				•	

Table 1.1: The data sets used in this thesis.

pants eventually joining the discussion. These results highlight the difficulty in predicting who will join a potentially concerning discussion based solely on their social connections, suggesting that false positives could indeed be frequent. Social media monitoring tools could therefore be incorrectly flagging people in large quantities, thereby drawing law enforcement attention away from truly hostile actors and towards unmalicious civilians.

Though hindsight showed that many new discussion participants had already been in the neighborhoods of existing participants, that did not translate into having accurate foresight into who would join a discussion in the future. This dichotomy lends credence both to the monitoring tools and their critics, meaning that government monitoring of social media could benefit greatly from further research on network forensics. Additional attention on this area could help law enforcement correctly allocate resources to monitor real threats and reduce the potential for impinging on freedom of expression.

1.7 Data

To sufficiently demonstrate the applicability of this research to broad varieties of social networks, this thesis relies on several data sets: one reproducing messages posted to an online forum, two collected from Twitter, two derived from police files, and one depicting rivalries between terrorist groups. In addition, hundreds of stylized social networks were generated for simulated experiments. Table 1.1 summarizes the use of these data sets throughout the thesis.

1.7.1 Real Networks

Islamic Network Online Forum

The Islamic Network was an online forum active from 2004 to 2010 that featured discussions on a variety of issues related to Islam and Muslim life. The 91,874 messages authored by 2,082 users of this largely English-language forum were published by the University of Arizona Artificial Intelligence Laboratory [197] in 2012. Though the Islamic Network was not an extremist forum by its nature, some of the messages appear to have had extremist sympathies. In addition to being

used in the aforementioned evaluation of the approaches employed by social media monitoring tools, these data were used in the evaluation of link inference algorithms described in Chapter 3. With the exception of posts that may have been deleted prior to the collection of this data set, it may be considered fully complete for the covered time period.

2020 US Election Tweets

The 2020 US elections featured several contested Senate races along with the competitive presidential election. Naturally, this was a popular topic of conversation on Twitter, from which a set of more than 541 million tweets was collected from December 2019 to February 2021 [1]. These tweets were used for the work presented in Chapter 2 studying abnormal bursts in account creations. Since they were collected using the Twitter streaming API [195], these tweets represented a one-percent non-random [140] sample of all tweets matching the collection terms listed in Appendix A.

2020 French Terror Tweets

In October 2020, France experienced two terrorist attacks with the beheading of Samuel Paty in Paris and the fatal stabbings of three people in Nice. A sample of more than 600,000 tweets covering these topics was collected in real time in October and November 2020 [25] using Twitter's streaming API [195]. For this thesis, the Twitter Academic Research API [194] was then used to collect the timelines of one thousand accounts that expressed anti-France or extremist sentiment in the initial sample, providing a fuller record of the messages they published on the platform. This pair of complementary collections was used in the link inference evaluation in Chapter 3 and in the exploration of how to improve downstream network analysis in Chapter 5. For the one thousand most active anti-France users analyzed in Chapters 3 and 5, approximately 68.9% of the tweets in the streamed sample also appeared in the collected timelines. This lack of completeness is due to some tweets having been deleted or made private prior to the timeline collection. However, the number of tweets in the original streamed sample was approximately 1.5% the number of tweets in the combined data, meaning that the timelines provided a much more comprehensive view of these users' activities.

Belo Horizonte Co-Offenses

In 2023, Toledo et al. [189] published a multiplex network indicating co-offenses between members of a Belo Horizonte criminal network, which was extracted from the files of military police in the Brazilian state of Minas Gerais. This network was used to help evaluate link inference algorithms in Chapter 3. Given that this particular criminal network made national news headlines [189], it likely received additional police attention that might make the data more comprehensive than police files on other criminals. However, the completeness of this information is not known, and it likely still suffers from the spotlight effect [41] by over-representing the connections of certain individuals.

[1] This data collection was first started by Dr. David Beskow.

Operazione Montagna Surveillance

A particularly valuable data set for covert network analysis originates from Italian police, who as a part of Operazione Montagna ("Operation Mountain") monitored several members of a Sicilian mafia through both in-person surveillance and telephone wiretapping [45, 76]. This bilayer network was used in the link inference evaluation described in Chapter 3. Since these networks were derived from information published in pre-trial detention orders [45], they may not contain all connections known to police at the time. However, Cavallaro et al. [45] noted that the court order related Operazione Montagna contained more wiretaps and stakeouts than other orders. Overall, the completeness of this network is unknown, and it likely suffers from the spotlight effect [41].

ViNSAR Terrorist Rivalries

The Violent Non-State Actor Rivalry (ViNSAR) project maps the rivalries between more than two hundred terrorist groups in Africa and Asia for the period spanning from 1990 to 2015 [51]. These dynamic data were separated based on the type of rivalry and used to evaluate different link inference algorithms, as described in Chapter 3. Collected manually from large databases of international news as well as reports from governmental and non-governmental organizations, the ViNSAR data are unlikely to be complete, particularly since prior research has shown that such databases are more likely to under-represent terrorist activity in areas with an unfree press [65] or inferior communication technology [64, 206]. This network is therefore likely to over-represent the connections of certain groups, though it is unclear which types of groups are more likely to be over-represented. The level of complteness of these data is also unclear.

1.7.2 Stylized Networks

Several of the experiments presented in this thesis made use of simulated social networks generated in three styles: small-world, scale-free, and uniformly random. All of these stylized networks were both weighted and directed, which meant using specialized models for their creation. This process is described in Chapter 3, including how these models were used to create multidimensional (i.e., multiplex or dynamic multiplex) networks from stylized base networks. The networks were then also used in Chapter 4 for the evaluation of methods to compute confidence intervals for network metrics.

1.8 Outline

The next three chapters of this thesis evaluate three techniques that can be used in the forensic analysis of social networks. Chapter 2 presents a regression analysis of sudden spikes in Twitter account creations that ties these phenomena to indicators of potential coordination. Chapter 3 proposes an algorithm for inferring missing connections in multidimensional networks and measures its performance against that of other algorithms. Chapter 4 compares the abilities of different network models to bootstrap confidence intervals for metrics calculated on stylized networks

with missing edges. After discussing these three techniques, Chapter 5 then explores four different contexts in which these approaches could potentially improve downstream network analysis. Finally, Chapter 6 summarizes the implications of this research for practitioners in network science, law enforcement, and social media analysis.

Chapter 2

Forewarning Impending Coordination[1]

2.1 Introduction

2.1.1 Motivation

The most challenging circumstance for analyzing social network activity is perhaps when there is little-to-no network information available. This can often be the case when trying to identify or predict activity as soon as possible, such as to prevent harm from taking place. In such instances, a viable strategy can be to identify anomalies in what little data can be attained. This chapter demonstrates a successful application of this approach to help counter the global threat to democracy presented by the orchestrated manipulation of social media [20], concern over which has only grown since the discovery of online influence campaigns carried out during the 2016 US presidential election [89].

Several studies have underscored the need to intervene early when countering the spread of online misinformation, since late responses can decrease people's belief in true information [50] and delayed corrections can struggle to reach wide audiences [202]. In contrast, nudging users to consider the accuracy of what they post can reduce misinformation sharing [104], and platform policies of removing deceptive content and accounts can lessen the influence of anti-vaccine groups [26].

Efforts to detect online information operations have largely focused on identifying automated accounts [24, 57] and false narratives [164, 176], but more recent research has recognized that coordination between accounts is critical to how influence campaigns manipulate platforms' recommendation algorithms [10, 32, 55, 151]. Indeed, since the operators of online information operations constantly change their tactics, efforts to identify individual malicious accounts or messages eventually fall into a cat-and-mouse game with hostile actors [55, 130, 151]. However, the basic goal of coordinating to manipulate social media recommendation algorithms remains the same [55], leading researchers to call for the development of methods for identifying the coordinated activity that propels online influence campaigns [55, 151, 178]. Simultaneously, other research has demonstrated the benefits of combining interoperable tools for identifying and studying information operations, since the fusion of multiple approaches can make analy-

[1]Parts of this chapter were published in the *Journal of Big Data* [13]. The article is available on request.

sis more comprehensive and allows for the triangulation of conclusions from multiple pieces of evidence [198].

Given the need to intervene against misinformation as early as possible, the call for identifying malicious coordinated activity rather than individual actors, and the desire to foster interoperable pipelines of relevant approaches, anomaly detection could be used to flag new accounts that may coordinate maliciously in the future. Since coordinated influence campaigns rely on groups of multiple accounts, the creation of such accounts within a short period may be detectable by social media platforms. A few analyst reports of online information operations have indeed documented the use of accounts that had been created together [70, 147, 196], and both Twitter [196] and Facebook [70] have stated that they suspend thousands of inauthentic new accounts, "often minutes after creation" [70, page 35]. However, the companies have not described how their systems operate.

Only a few researchers have systematically analyzed bursts of account creations. Takacs and McCulloh [185] discovered bursts of created accounts that followed 2018 US Senate candidates and showed that they were statistically different from older accounts in terms of properties such as numbers of followers. However, the behavior of these accounts was not examined. In a different study, Jones [106] found spikes in monthly account creations among Twitter users discussing Qatar and manually reviewed a sample of ten thousand tweets and corresponding author profiles, therefrom identifying large proportions of bots. However, the coding and classification process used in this study was not clear, statistical testing of the results was not undertaken, and the data's network structure was not considered. This makes detecting anomalies in account creations a promising direction for studying how to provide early warnings for future coordination on social media.

2.1.2 Objectives

This chapter investigates how sudden spikes in online account creations might warn of potentially nefarious future coordination. Aggregating the creation dates of accounts featured in more than 541 million tweets discussing the 2020 US elections led to the discovery of several suspicious bursts of account creations. This research statistically analyzes these data to explore the following questions.

1. How do accounts created together in a burst compare to each other in terms of future behavior?
2. How does the political engagement of such accounts compare to that of accounts not created during bursts?
3. Are spikes in account creations associated with potentially malicious characteristics, such as the presence of bots or the sharing of misinformation?

In answering these questions, this chapter relies on regression models relating account creation bursts to various indicators of potential coordinated or concerning activity.

2.2 Methods

2.2.1 Data Collection

To investigate potential coordination during the 2020 US elections, the Twitter streaming API was used to collect election-related tweets from 1 December 2019 to 17 February 2021 [2]. This collection initially followed accounts and hashtags associated with the Democratic presidential primaries but was later modified to cover users and topics relevant to post-primary presidential campaigning and to particularly contested Senate races. Some time after the election, the collection was changed to focus on the presidential inauguration. More specific details are available in Appendix A. In total, the collection amassed 541,413,669 tweets published by 21,114,981 users. Though a larger set of tweets on the 2020 elections exists [48], related publications have thus far only analyzed smaller subsets of that collection [46, 75, 175], thereby making this study of account creations the largest known analysis of tweets on the 2020 US elections.

2.2.2 Network Construction

Following the data collection, the directed communication network between users was derived from the collected tweets. Each edge in this network was weighted by the number of tweets published by the source user that targeted the other account by retweeting, quoting, mentioning, or replying to that account. Additionally, the hashtag usage network was constructed by counting the number of tweets published by each account that contained a given hashtag. When creating both networks, the contents of a retweet were attributed to both the retweeted and retweeting accounts. In contrast, only the additional content of quote tweets was attributed to the quoting users, whereas the quoted content was attributed to the quoted account. This policy reflected the prevalence on Twitter of quoting a tweet even when expressing disagreement with it.

2.2.3 Burst Detection

Since a prior analysis of an information operation described accounts being created over the span of a few days [147], all account creation times were extracted from the election tweets and amalgamated by day. Since large social media platforms experience notable network effects in user growth during their early existence, the subsequent analysis was limited to accounts created in the ten years preceding the end of the data collection period.

Identifying sudden spikes in account creations was accomplished by viewing the histogram of daily account creations as a one-dimensional signal. This signal was run through the cell-averaging constant false alarm rate (CA CFAR) algorithm, which indicates a detection when the signal exceeds a threshold computed using nearby signal values and a sensitivity parameter known as the probability of false alarm (PFA) [165]. More specifically, the threshold for a given day was calculated by applying a constant factor involving the PFA to the average daily number of accounts created during a training window centered on the specified day, not counting the day itself and a pre-specified number of days immediately adjacent to it (known as guard cells).

[2]This data collection was first started by Dr. David Beskow.

This research employed a training window encompassing the thirty days before and thirty days after a given date, with the exception of the week centered on that date. In other words, the three days immediately adjacent to each day were designated as guard cells while the remaining fifty-four days were used to calculate the detection threshold. This process has the notable limitation of relying on information from after a given date, but this research was intended as a proof of concept to be used as a starting point for future work. To summarize, Equation 2.1 defines the calculation of the CA CFAR threshold [165] for day d, in which $\boldsymbol{h}$ is the vector of daily account creations, $g = 3$ is the number of guard cells on each side, and $t = 27$ is the number of training cells on each side.

$$t_d = \sum_{i=g+1}^{g+t} (\boldsymbol{h}_{d-i} + \boldsymbol{h}_{d+i}) \left({p_{fa}}^{\left(-\frac{1}{2t}\right)} - 1 \right) \tag{2.1}$$

Since the sensitivity parameter could impact results, three PFA values were chosen manually to highlight three levels of bursts in account creations: particularly tall ones (PFA = 0.25), shorter ones that nevertheless still appear as spikes (PFA = 0.30), and very short ones that could also qualify as noise (PFA = 0.35). This resulted in 669 bursts detected at a PFA of 0.35, 112 spikes identified with a PFA of 0.30, and 51 bursts detected with a PFA of 0.25. Figure 2.1 indicates that the medium sensitivity yielded detections that were likely to be useful, whereas the lowest sensitivity identified spikes that may have been impacted by world events. It is also important to note that the bursts detected at higher sensitivities were supersets of those identified using lower sensitivities.

2.2.4 Regression Analysis

After identifying sudden spikes in account creations, regression was used to determine whether these bursts were tied to phenomena that could indicate coordination. These phenomena were quantified as dependent variables calculated based on all the accounts created on a given date, meaning the level of analysis was a day. Since the independent and control variables were also defined for each day, modeling the data at the daily level maintained statistical independence between data points. All dependent variables were defined as ranging from zero to one, so one-part fractional regression models [153] were implemented as generalized linear models with quasi-binomial families and logit link functions [163]. A separate model for each dependent variable was fit to a binary independent variable indicating whether each data point was associated with a burst or non-burst day. Moreover, since the account creation times were collected from discussions of the 2020 elections, there was a notable bias toward newer accounts that can be seen in Figure 2.1. The age of each data point was therefore added to the regression models to control for these effects.

After fitting each model, z-tests were performed for each variable coefficient using robust standard errors [207]. The average marginal effect (AME) of the binary burst variable was then calculated to determine the magnitude of its relationship to the various phenomena of interest. This was done by averaging the differences between model predictions after assigning all data points as bursts and after assigning them all as non-bursts [88]. Subsequently, z-tests were performed on the AMEs with robust standard errors estimated via the delta method [88].

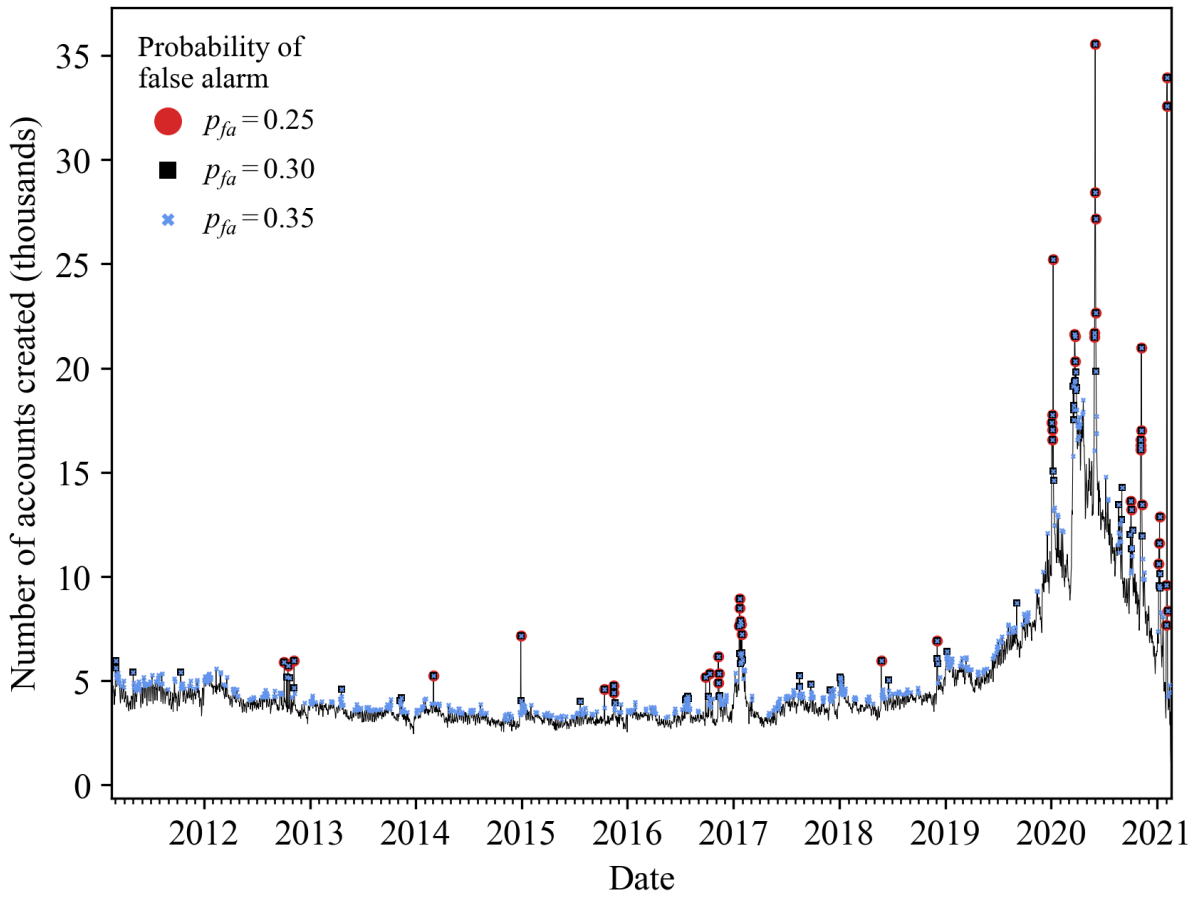


Figure 2.1: The number of accounts created each day that later appeared in the collection of election-related tweets. Spikes detected using different CFAR sensitivities are indicated by different symbols.

To make the results more interpretable, each AME was normalized by the standard deviation of the corresponding dependent variable. This also meant that the AMEs from models for dependent variables with natural tendencies to be particularly small (such as network densities) could be directly compared to the AMEs from other models. Finally, since this analysis required one regression model to be fit for each combination of PFA value and dependent variable, the Benjamini-Hochberg procedure [19] was applied to all z-test p-values to control for a false discovery rate of five percent.

2.2.5 Measured Features

Connection Similarity

The first quantity measured to identify potential coordination was the similarity of the users with whom accounts chose to communicate. Specifically, this was measured from the communication network as the average cosine similarity of the neighborhoods for accounts created on the same

day. The average local clustering coefficient [205] was also calculated for accounts created on the same day, which indicated the propensity for these accounts' neighbors to communicate among themselves. The mean clustering coefficients ($n = 3654$, $\mu = 0.390$, $\sigma = 0.015$) and mean neighborhood cosine similarities ($n = 3654$, $\mu = 0.265$, $\sigma = 0.036$) served as dependent variables for separate regression models.

Density of Contemporaries

A natural indicator of coordination between accounts created on the same day would be if these contemporaries ended up communicating with each other more often than would normally be expected. The density of communication between accounts created on a given day was therefore measured from the full communication network. Regression models were then fit to the resulting density values ($n = 3654$, $\mu = 1\text{e}{-06}$, $\sigma = 2\text{e}{-06}$).

Hashtag Similarity

Coordinating accounts may also discuss more similar topics, so the hashtag usage network was used to calculate the mean cosine similarity of hashtag choices for the accounts created on a given day. These daily hashtag similarities ($n = 3654$, $\mu = 0.007$, $\sigma = 0.005$) then served as a dependent variable in the regression analysis.

User Stance

Another potential indicator of coordination among users would be if they agreed more than expected on political issues that were normally controversial. The 2020 US elections featured several such issues, including the mail-in voting and mask-wearing regulations instituted in response to the COVID-19 pandemic [11, 143]. Indeed, Figure 2.2 reveals that usage of partisan hashtags on these topics decreased around election day before partisan discussion of mail-in voting then spiked again on the day the election was called. These partisan hashtags, which are listed in Appendix A, were used to seed the stance detection algorithm built into the ORA-PRO software [44], which starts by labeling users with the initial set of hashtags and then employs influence propagation across both the communication network and the hashtag usage network to label the stances of other users and hashtags [119]. Running this algorithm on the US election networks yielded stance classifications for each user for each of the two topics, along with corresponding confidence values. These confidence values were linearly remapped and combined with the stance classifications to range from zero (against) to one (in favor), with 0.5 representing a neutral stance. Finally, disagreement among accounts created together on a given day was measured by computing the standard deviation in their stances for each of the two topics. These standard deviations in mail-in voting stances ($n = 3654$, $\mu = 0.283$, $\sigma = 0.006$) and mask wearing stances ($n = 3654$, $\mu = 0.263$, $\sigma = 0.006$) were used as dependent variables in the regression analysis.

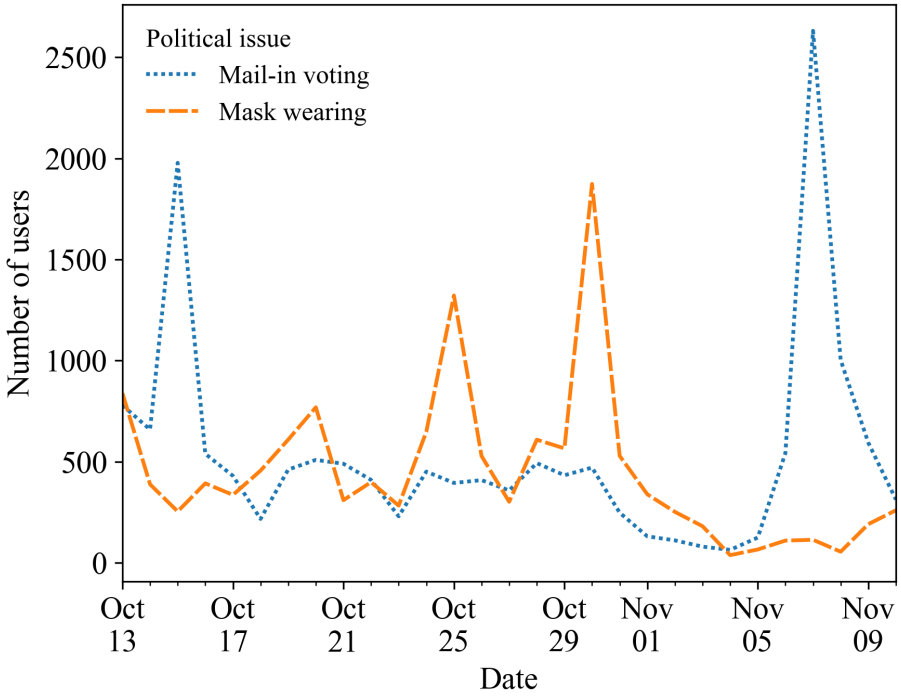


Figure 2.2: The daily numbers of accounts mentioning partisan hashtags about mail-in voting and mask wearing in late 2020. These hashtags are listed in Appendix A.

Hashtag Synchrony

To examine whether accounts created together in bursts may have coordinated their activity in time as well as content, part of the methodology employed by Ng and Carley [145] was adopted by using the ORA-PRO software [44] to identify synchrony in hashtag usage when discussing controversial political issues. More specifically, all tweets that used any of the partisan postal-voting or mask-wearing hashtags were first isolated, and ORA-PRO was then leveraged to identify instances in which tweets containing the same hashtag were published within five minutes of each other, which was the temporal threshold advocated by Ng and Carley [145]. For each day, these results were used to count the number of pairs of created accounts that ended up exhibiting synchronized hashtag usage at least once in the entire data set. These counts were then converted to proportions by dividing by the number of possible pairs of participating accounts created each day. Since not all users participated in partisan discussions of mail-in voting or mask wearing, this also resulted in not all dates in the ten-year period having an associated proportion of synchronized accounts. Only the valid dates were included in the corresponding regression models, in which the proportions of synchronized accounts ($n = 3651$, $\mu = 4\text{e}{-}04$, $\sigma = 0.001$) served as a dependent variable.

Bot Activity

Online influence operations have made extensive use of automated social media accounts [55, 74], meaning that their presence can be an indicator of potentially malicious activity. The Tier-1

BotHunter model, which applies a random forest regressor to several features including tweet content and timing [24], was used to estimate the probability that each account in the election tweets was a bot. Following prior research on making more stable bot classifications using limited numbers of tweets per account [146], a probability threshold of 70% was applied to classify accounts as bots or non-bots. These bot classifications were then used to calculate the proportion of accounts created on a given day that were bots, and the resulting proportions ($n = 3654$, $\mu = 0.152$, $\sigma = 0.052$) served as a dependent variable for the regression analysis.

Sharing of Low-Credibility Sources

Since not all online coordination is malicious, the likelihood that accounts would share links to low-credibility sites was also of interest. All URLs in the collected tweets were therefore extracted and expanded [215] before then identifying their top-level domains. These domains were classified as low-credibility if they belonged to an inexhaustive list of 954 Web domains that were either (a) included as black sites in the misinformation study by Grinberg et al. [90], (b) listed as fake news by PolitiFact [160], or (c) rated by Media Bias/Fact Check as being conspiracy or questionable sites with low or very low factual reporting [132, 133]. This list of domains can be found in Appendix A. These URL classifications were then used to identify which accounts tweeted or retweeted at least one low-credibility link in the US election data. Finally, this information was used to compute the proportion of accounts created on each day that ended up sharing low-credibility links. The resulting proportions ($n = 3654$, $\mu = 0.024$, $\sigma = 0.006$) were then employed in the regression analysis.

2.3 Results

2.3.1 Communication with Others

The regression analysis found that users' communication in the 2020 election data was more similar for accounts created together during a burst compared to accounts created during other periods. The fractional regression models showed that even after controlling for account age, average neighborhood similarity was associated with significant positive coefficients for the variable indicating that a day exhibited a spike in account creations. Table 2.1 shows the AMEs for the burst indicator variable from each of the regression models, with complete model specifications and results available in Appendix B. For spikes detected at medium sensitivity, average neighborhood similarities increased by 1.11 standard deviations for accounts created during bursts, and this result was even stronger (1.38 standard deviations) for spikes detected with the lowest sensitivity. This revealed greater homogeneity in the communication patterns of accounts created during bursts, even when controlling for their account age.

Separate regression models also showed that the communities around burst accounts tended to be more tightly knit, with their average clustering coefficient increasing by 0.51 standard deviations for medium spikes and 0.76 standard deviations for particularly tall spikes. This indicates that accounts created during bursts were more strongly embedded in their communities, which could have been echo chambers or bot nets.

PFA	0.25		0.30		0.35	
Dependent Variable	AME	$\frac{\text{AME}}{\sigma}$	AME	$\frac{\text{AME}}{\sigma}$	AME	$\frac{\text{AME}}{\sigma}$
Mean neighborhood similarity	0.050^{***}	1.38	0.040^{***}	1.11	0.017^{***}	0.46
Mean clustering coefficient	0.011^{*}	0.76	0.007^{**}	0.51	$9e{-}04^{\dagger}$	0.06
Density amongst contemporaries	$1e{-}06^{**}$	0.62	$9e{-}07^{***}$	0.42	$3e{-}07^{***}$	0.15
Mean hashtag similarity	$0.010^{\dagger}$	2.11	$0.005^{\dagger}$	0.97	$9e{-}04^{*}$	0.19
Mail-in voting stance std. dev.	-0.016^{***}	-2.55	-0.009^{***}	-1.42	-0.002^{***}	-0.34
Mask wearing stance std. dev.	-0.011^{**}	-1.84	-0.005^{*}	-0.83	-0.001^{*}	-0.18
Voting + masking synchronization	$3e{-}04^{\dagger}$	0.21	$2e{-}04^{\dagger}$	0.13	$1e{-}04^{\dagger}$	0.08
Prop. bots $\geq 70\%$	0.022^{***}	0.41	0.016^{***}	0.31	$3e{-}04^{\dagger}$	$6e{-}03$
Prop. sharing low-credibility sites	$0.003^{\dagger}$	0.53	0.004^{**}	0.73	0.002^{***}	0.41

Table 2.1: The average marginal effects (AMEs) and associated corrected p-values for the binary burst variable from models fit to each measured behavioral feature. Each AME was standardized by dividing by the standard deviation (σ) of the dependent variable. Complete model specifications and results are available in Appendix B. Significance: † for $p < 0.1$, * for $p < 0.05$, ** for $p < 0.01$, and *** for $p < 0.001$.

2.3.2 Talk Among Contemporaries

The regression results also revealed that the density of connections tended to be greater among accounts created during bursts, meaning they were more likely to communicate among themselves. Accounts created during bursts exhibited communication densities that were 0.42 standard deviations higher when examining medium-sensitivity spikes and 0.62 standard deviations higher for low-sensitivity spikes. Figure 2.3 shows an example of the greater communication density that tended to exist between burst accounts. This greater communication density suggests that burst accounts could have been coordinating their activity to, for example, construct echo chambers or prop up specific accounts.

2.3.3 Hashtag Usage Patterns

Accounts created during medium or tall spikes appeared to use more similar sets of hashtags, but this result was not significant after correcting for the false discovery rate. The mean cosine similarity of hashtag usage between accounts created during the same day tended to be almost one standard deviation higher for medium burst days and over twice as large for the tallest spikes,

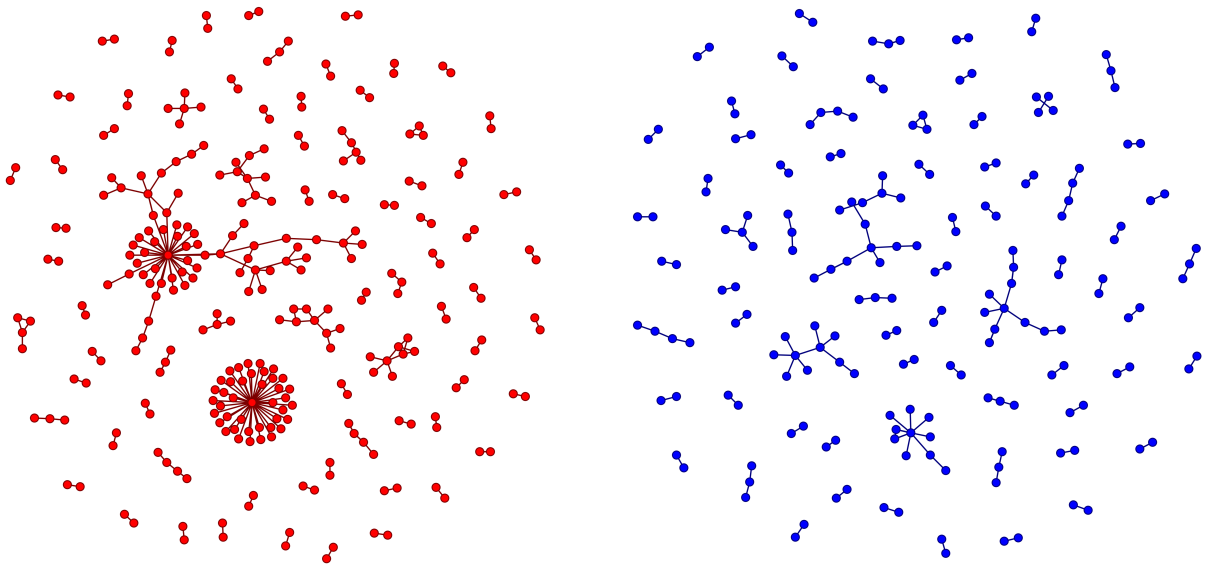


Figure 2.3: Example communication networks for two days with similar numbers of created accounts: 6 January 2020 (left), identified as a burst at the lowest sensitivity, and 24 April 2020 (right), a day not identified as a burst at any sensitivity. Not shown are 16,261 accounts from 6 January and 16,089 accounts from 24 April who did not communicate with any other accounts created on their respective days.

but the corresponding p-value was no longer below 0.05 after correction.

2.3.4 Agreement on Controversial Issues

For both mail-in voting and mask wearing, accounts created during bursts tended to exhibit significantly less disagreement among themselves. On mail-in voting, disagreement fell by 1.41 standard deviations on average for medium spikes and 2.55 standard deviations for the tallest spikes. On mask wearing, disagreement similarly decreased by 0.83 standard deviations for accounts created during medium-sensitivity bursts and 1.84 standard deviations for accounts created during low-sensitivity bursts. Spikes in account creations were therefore tied to disproportionately less disagreement on political issues that should have been controversial, contributing to the suspicion that these spikes were partly caused by coordinated activity.

2.3.5 Synchronous Behavior

Within partisan discussions of mail-in voting and mask wearing, the proportion of contemporaries that ended up synchronizing their hashtag choices was not significantly related to those accounts being created during bursts. The results were close to being significant, with $p < 0.06$ for the medium bursts, suggesting that a different approach (such as studying the entire data set in this manner) might have been able to discover a different kind of synchronization.

2.3.6 Bot Activity

Account creation bursts also tended to see slightly more bots being created. During medium spikes, an average of 1.6% more bots were created, equivalent to an increase of 0.31 standard deviations. The tallest spikes exhibited an average of 2.2% more bots, which represented an increase of 0.41 standard deviations. These effect sizes were significant but smaller than for the other results, which could have been impacted by the particularly conservative probability threshold used to more stably classify bot accounts. Though this means that the accounts labeled as bots were more likely to actually be automated, it is also more likely that the classification missed some automated accounts.

2.3.7 Sharing of Low-Credibility Sources

Though the rest of the regression analysis overwhelmingly pointed to burst accounts exhibiting more similar behavior than accounts created during other periods, this does not show that the added potential for coordination was harmful. However, the regression results also revealed that accounts created during medium bursts were significantly more likely to eventually share links to low-credibility sites, with the likelihood of sharing such links increasing by 0.73 standard deviations. After correcting for the false discovery rate, this result was not statistically significant for low-sensitivity bursts. This supports the hypothesis that the tallest spikes could have been organically influenced by world events. Reports on influence campaigns have described coordinated networks of hundreds of accounts [70, 147, 196], whereas the lowest-sensitivity bursts involved increases of many thousands of accounts. Indeed, ten of the tallest spikes occurred on days following the US elections of 2012, 2016, and 2020. Account creation bursts were therefore only significantly tied to greater sharing of low-credibility sources if the spikes were not so large that they would have required sizable authentic activity. Nonetheless, it is clear that the bursts that were likely to have been created by inauthentic activity were associated with greater sharing of low-credibility sites.

2.4 Discussion

This analysis provides statistical evidence for the relationship between abnormal bursts of Twitter account creations and indicators of potential coordination. Accounts created during these bursts communicated with more similar sets of users, were embedded in more tightly connected communities, and communicated more than expected with other accounts created on the same day. They also tended to agree more on controversial political issues, were slightly more likely to be bots, and tended to share more links to low-credibility sites. Though the coordination indicators based on hashtag choices and their timings were not significantly related to spikes in account creations, this work presents ample substantiation of the suspicion that social media accounts created *en masse* could potentially warn of future coordinated influence operations.

Ideally, social media companies would identify and thwart malicious actors as soon as possible. However, such rapid decision-making naturally comes at the expense of having less information available when deciding whether to intervene. This is of particularly serious concern

given that social media interventions can be used to disrupt legitimate forms of coordination like protests [62]. When available information is insufficient to confidently decide to suspend an account but the actor still appears to be suspicious, "softer" measures for countering possible influence campaigns should be preferred.

This work has shown how the very act of online account creation can draw suspicion upon sets of possibly malicious actors. Accounts created in unnatural bursts were tied to increased similarity in behavior and political stance as well as greater likelihoods of being automated and sharing links to low-credibility sites. Detecting these bursts to swiftly identify suspicious accounts could therefore assist social media platforms in pre-emptively hindering the dissemination of misinformation, which can be extremely valuable considering the rapidity with which it spreads [201]. However, like with previous work examining suspicious account creations [121], the burst detection explored in this work cannot be used in isolation because of the legitimate accounts that would be caught in the crossfire.

To keep pace with the constant changes in malicious actors' online strategies, scholars and professionals would be best served by developing interoperable tools and methodologies [198] that can be used by social media companies to gradually build evidence of harmfulness against actors and incrementally hamper their efforts as these suspicions are substantiated. Just as countries can sanction foreign entities to varying levels of severity, so too can social media platforms incrementally sanction bad actors in ways other than outright suspension. However, published research has focused largely on developing individual detection systems with little effort to integrate them [198].

Since the companies themselves control the algorithms used to recommend content to their platforms' users, they are free to penalize certain content in these recommendation systems. Social media companies are obviously aware of this, already implementing policies to lower the rankings of false or undesirable messages in their recommendation algorithms [69, 193]. Facebook has said it sometimes goes as far as to apply this "deranking" to all content from a user who has repeatedly shared false content, and Twitter stated that it tracks whether "the same person signs up for multiple accounts simultaneously" when deciding whether to decrease a user's ranking in its recommendation systems [193]. Yet despite these platform policies and additional systems for automatically detecting and closing inauthentic accounts [70, 196], this research shows that abnormal spikes in account creations were still identifiable in published Twitter data and that they were associated with phenomena suggesting potentially malicious coordination.

By detecting sudden bursts of account creations, social media companies have the opportunity to immediately penalize and investigate questionable accounts until a final determination can be made to suspend them or clear them of suspicion. These companies could further improve this strategy by implementing and investigating more advanced burst detection strategies. In concert with evidence produced by other tools and methods for detecting misinformation campaigns, the identification of account creation bursts could thereby help mitigate the effects of future attacks on democratic societies. Given the promising potential of integrating multiple anti-misinformation methods for countering malicious actors' constantly changing strategies, researchers should strive to develop more integrable methods for identifying bad actors and harmful content on social media.

2.5 Conclusion

2.5.1 Limitations

Though this research ties account creation bursts to the potential for future coordination, solely detecting spikes in account creations cannot be relied upon to definitively ban accounts, since many genuine users would be penalized. This strategy must be used in conjunction with other methods for monitoring social media accounts and identifying coordinated or malicious activity. Moreover, this work only studied a sample of Twitter conversations discussing the 2020 US elections, which means these results may not generalize to all Twitter activity or to other social media platforms. Indeed, the fact that the data were collected largely by following specific hashtags may have interacted with the methodology to impact its results, such as the lack of statistical significance for the indicators involving hashtag choices and the timing of uses of those hashtags. Finally, social media platforms do not publicize their inner workings and can change rapidly, so it is uncertain how these results were influenced by Twitter's algorithms (e.g., for choosing which tweets to serve in the data collection process) or how they would change after a significant amount of time has passed.

2.5.2 Contributions

Using the largest fully analyzed Twitter data set on the 2020 US elections to date [13], this work shows that abnormal spikes in the daily numbers of Twitter account creations were associated with heightened indicators of possibly malicious coordination. Compared to other accounts, those created during these bursts exhibited more similar behavior, agreed more on normally controversial issues, were more likely to be bots, and were more likely to share links to low-credibility sites. Though classifying accounts based solely on their creation date is not a viable way of conclusively determining whether to suspend them, social media companies could pre-emptively limit their influence by incorporating this strategy into existing platform policies for lowering the ranking of certain content [69, 193]. If combined with other methods for detecting coordinated malicious activity [198], tracking account creation bursts would help platforms hinder coordinated influence campaigns from their very beginnings, even before inauthentic accounts display undesirable behavior. Given the importance of stopping malicious content as early as possible, this would contribute to effectively countering online information operations.

By studying the creation times of online accounts, this research also demonstrates how detecting anomalies in extremely scarce information can help reveal hidden activities and motivations in network-based systems like social media platforms. Anomalies in account creation times were predictive of future network behaviors, making this a viable approach for supplementing social network analysis when data are particularly scarce. Indeed, many other kinds of anomalies are worth exploring and would also contribute to the forensic analysis of social networks.

Chapter 3

Emending Multidimensional Networks[1]

3.1 Introduction

3.1.1 Motivation

Aside from collecting better data, the most obvious remedy to imperfect network data is to use what is already known about the network to fill in missing connections and remove spurious ones. The task of revealing hidden links, known as link prediction or link inference, has spawned numerous algorithms for finding such edges [118, 128], some of which have also been adapted for identifying ties that should be discarded [91, 152, 161, 167, 218, 220]. However, very few of these methods operate on networks that are both dynamic and multiplex. Moreover, the small number of algorithms developed for such networks were all intended for predicting future connections and can be computationally expensive [36, 120, 212], require significant tuning [36, 150, 212], or demand specific kinds of data [2, 209]. This makes it challenging to improve the quality of the rich complex networks that are evermore commonplace in the era of big data [102], such as those characterizing social media, epidemiological spread, transportation systems, and more [58].

Concerningly, link inference algorithms have also largely been evaluated in potentially unrealistic settings. Most studies have relied on evaluations that take uniformly random sub-samples of a network's connections and subsequently test whether an algorithm can identify the deleted links [214]. Though such uniformly random procedures are helpful for providing theoretical bounds on performance when edges are missing completely at random (MCAR) [126, 166, 169], they are unlikely to reflect the real processes that cause connections to be missing in an observed network [214], such as when edges are missing not at random (MNAR) [126, 166, 169]. Police data, for example, are known to suffer from non-random "spotlight" effects because of their focus on specific individuals [41]. Link inference algorithms should therefore be tested under scenarios that are more likely to capture the real-world causes of observation error when collecting complex network data. Moreover, network analysts frequently deal with weighted network information, but the consequent intricacies have not been adequately considered by prior link

[1]Parts of this chapter have been reproduced from a manuscript being drafted for publication. The draft article is available on request. Parts of this chapter were also presented at the 2024 and 2025 Informed Democracy and Social-cybersecurity Conferences [15, 17].

inference studies. Whereas prior research has largely focused on the inference of connections that are completely missing from an observed network, analysts studying weighted networks may be interested in identifying the ties with weights that are too low when compared to reality, particularly since this information can be important when identifying the central actors in a network.

Network scientists can also find themselves working with networks that contain edges that do not exist in reality. The identification of such spurious links has received significantly less attention than the identification of missing links [167, 217, 220], with no known methods operating on networks that are both dynamic and multiplex. However, a small number of studies applying link inference methods to the identification of spurious links [91, 152, 161, 218, 220] shows promise for expanding this area of research. Since link inference algorithms are often designed to compute a likelihood score for the existence of each edge, the lowest-scoring edges can be regarded as the ones most likely to be spurious. Indeed, this approach has been tested using link inference algorithms relying on stochastic block modeling [91], information diffusion [218], structural Hamiltonians [152], force-based community detection [161], and classic link inference methods [220]. However, identifying spurious ties via link inference has not been attempted on dynamic networks, and only Samei and Jalili [167] have attempted it on multiplex networks. No one has evaluated the use of link inference to identify spurious links in dynamic multiplex networks.

3.1.2 Objectives

This chapter proposes a flexible algorithm for identifying missing or spurious links in multidimensional networks and evaluates its performance on both stylized and real networks, with the experiments on real networks being designed to mimic practical situations using covert network data. Specifically, this research answers the following questions.

1. How can an algorithm discover missing and spurious connections in multidimensional networks?
2. How well does this algorithm perform in realistic scenarios involving covert networks?
3. How do network characteristics impact the algorithm's performance?

In answering these questions, this chapter compares the proposed algorithm to nine other link inference algorithms in evaluation trials conducted on artificial stylized networks and real covert networks.

3.2 Methods

3.2.1 Tensor Perturbation Method

Since dynamic and multiplex networks can be represented as tensors [58, 67], the proposed tensor perturbation method (TPM) has been designed to correct the entries of a network's tensor representation, thereby allowing investigators to format their data to fit their situation. For example, social media posts could be encoded as a four-dimensional tensor, with each dimension representing the post authors, the users mentioned in the posts, the hashtags used, and the timing

of the posts, respectively. In such a tensor, the value at position (i, j, k, l) would be the number of posts authored by account i during time period l that mentioned user j and contained hashtag k. Running TPM on this tensor would infer the missing messages between accounts, estimate when they may have been posted, and surmise what hashtags may have been included. This presents a significant advantage beyond other dynamic multiplex link inference algorithms, which assess the overall proximity between actors without dissecting that proximity into different link types.

Appendix C describes TPM in detail. At its core, the algorithm learns the fundamental structure of a weighted, directed network by extending a matrix-based approach by Wang et al. [204] (hereafter referred to as NMFP) to tensors. This occurs by repeatedly perturbing the input tensor and averaging together the resulting reconstructions. First, noise is added to the network in the form of inserted or removed connections, thereby simulating the presence of erroneous links or the absence of real links. [2] Second, the altered tensor is decomposed into its latent factors (i.e., connection patterns) and then reconstructed as a low-rank approximation. This second step essentially compresses the altered tensor to learn its structure and remove the added noise. By repeating these steps several times and averaging together the reconstructed tensors, TPM yields a final approximation of the true network, including the potentially missing links.

The second perturbation step reconstructs an approximation of the noisy tensor via incomplete non-negative Tucker decomposition, an analogue to singular value decomposition (SVD) that factorizes a tensor into a set of factor matrices (one per dimension) and a core tensor containing coefficients for those factors [112]. Like with SVD, an incomplete decomposition only finds the most salient components for efficiently approximating the original data [112]. Unlike the canonical polyadic decomposition used in most tensor-based link inference algorithms [67, 80, 182, 213, 219], the incomplete Tucker decomposition can "compress" each dimension by a different ratio, enabling more adaptive learning of structural patterns and allowing the algorithm to not compress crucial dimensions, such as those corresponding to different layers or time slices. Constructing this low-rank approximation entails determining the optimal multilinear rank (i.e., the best numbers of factors) for approximating the tensor, a task for which several complex algorithms have been proposed [137]. TPM can instead follow the lead of NMFP [204] by using the Colibri-S algorithm [190] to select the rank with which to decompose an adjacency matrix, but the input tensor must first be unfolded [112] along each dimension before using Colibri-S to obtain the optimal rank in that dimension. Since the noise added to the input data should not change the optimal number of latent factors for representing the unseen true data, this process for finding the best multilinear rank only needs to be executed once on the raw input tensor, thereby saving on computational costs.

By repeating the perturbation steps and averaging the reconstructed tensors across several trials, TPM produces a tensor that scores the likelihood of each potential link. Further processing can also be performed, such as slicing along a certain dimension if an analyst only cares about a certain time period or link type. Alternatively, the tensor can be summed across dimensions to study links irrespective of their time or type. TPM's flexibility therefore frees analysts to choose their data structure and identify interesting relationships.

[2] For this work, ten percent of the integer edge weights were subtracted uniformly at random, but this is only one possible implementation. This neutral noise model made it possible to apply the same implementation to all data sets but introduced limitations that are discussed later in this chapter.

3.2.2 Benchmark Algorithms

The link inference performance of TPM was compared to that of several other methods, including three classic node similarity measures based on the shared neighborhood between two nodes: common neighbors (CN) [144], Adamic-Adar (AA) [1], and resource allocation (RA) [222]. A link inference method based on robust principal component analysis, the matrix completion (MC) method [155], was also tested. Additionally, TPM was evaluated against three methods that perturb a network and decompose its adjacency matrix, namely the structural perturbation method (SPM) [129], the matrix perturbation and decomposition (MPD) method [211], and the NMFP method by Wang et al. [204]. Finally, TPM was compared to the MLRW algorithm for multilayer networks based on random walks [141] and the NCP4 algorithm for dynamic networks based on tensor decomposition [219]. The parameters for each benchmark algorithm were optimized through five-fold cross-validation, in which each of the five trials proceeded by randomly removing ten percent of the input network's edges and measuring the algorithm's ability to recover those edges in terms of area under the precision-recall curve (AUPRC) [118, 204, 214, 221]. Appendix C provides more details on the optimization of these algorithms.

3.2.3 Performance Measurement

Link inference performance was measured using area under the receiver operating characteristic curve (AUROC), precision, and area under the precision-recall curve (AUPRC). Though AUROC is a popular performance metric in machine learning, several studies have highlighted that the extreme class imbalance and lack of a clear negative set in link inference problems can lead to deceiving results when using AUROC [125, 204, 214]. Many researchers have therefore moved to employing precision and AUPRC [118, 204, 214, 221]. Precision measures the identification of true positives as a proportion of the links in the testing set and is equivalent to the precision-at-k metric when k is the number of links in the testing set [54, 118, 214, 221]. Meanwhile, the AUPRC can summarize performance across various thresholds for the likelihood scores produced by link inference algorithms [214, 221].

Importantly, analysts studying imperfect networks often do not know precisely how many edges they are missing. Therefore, to assess link inference performance under different levels of data scarcity, the proportion of missing or spurious links was varied between evaluation trials. Furthermore, imperfect networks that are weighted can contain two kinds of missing or spurious edges: those that are completely erroneous (i.e., they have a weight of zero in either the observed or the true network but have nonzero weight in the other network) and those that exist in the observed network but have an inaccurate weight compared to reality. Since analysts' interest in these kinds of observation error may differ, the link inference performance measures were calculated twice: once considering both types of erroneous edges and again considering only the completely erroneous edges.

3.2.4 Inferring Simulated Missing Links

Generating Simulated Networks

The link inference performance of TPM was first tested in hundreds of simulated trails on multidimensional stylized networks of three types: uniformly random, scale-free, and small-world. Two sets of experiments were conducted: one on multiplex networks with three layers each and another on dynamic multiplex networks with three layers and three time periods. For each style, networks were generated consisting of fifty or one hundred nodes and densities of five percent or ten percent. In keeping with the goal of evaluating TPM for use on many kinds of complex networks, specialized models were used to generate multidimensional stylized networks that were both weighted and directed.

Each multidimensional stylized network was constructed by first generating a single-layer "base" network and therefrom deriving multiple network layers or time slices. Specifically, the edge weights in the base network were used to parameterize Poisson distributions from which the weights for each new network layer or time slice were then drawn. This approach allowed each network layer and time slice to vary while still being related to the underlying base network.

Generating the base networks required a different model for each style. Scale-free networks were constructed using the preferential attachment model developed by Pi et al. [159], which enabled the creation of weighted, directed networks. However, this attachment process was run without adding edges between existing nodes, making it more closely resemble the original Barabási–Albert model [5]. The model was further modified by rounding the final edge weights to the nearest integer value, since the models used for the other network styles produced integer weights. Finally, the model required the specification of a δ parameter, which dictated the rate at which edge weights increased when adding a new node. Two values were tested for this parameter: 0.5, which can represent situations like scientific collaboration in which a new collaboration does not significantly influence existing collaborations [12], and 1.0, which can represent cases such as transportation networks in which new activity can strain existing pathways [12]. Meanwhile, the initial number of nodes in the initial fully connected cluster was set to five percent of the total number of nodes in the network, and the number of new edges created during each attachment step was calculated based on the desired network density.

Small-world networks were constructed by modifying the directed Song and Wang [179] model to add the weighting mechanism proposed by Garlaschelli [81] for uniformly random networks. The Song and Wang [179] model defines the probability of existence for each directed edge in the small-world network based on a β parameter that determines the probability of "long-distance" edges that cut across the ring lattice of the small-world network. Following the Garlaschelli [81] proposal, the complement of each edge probability was used to parameterize a geometric distribution from which the final edge weight was drawn. To study the impact of the β parameter on link inference performance, both 0.1 and 0.2 were tested as parameter values. The base degree of the nodes in the lattice was calculated to produce a network of the desired density.

Uniformly random networks were constructed using the same model as the small-world networks by setting $\beta = 1$. Since the β parameter defined the probability of random "long-distance" edges, maximizing β in the Song and Wang [179] model makes it equivalent to the Garlaschelli [81] model for weighted Erdős–Rényi networks, except for the addition of edge directionality.

Simulating Missing Links

Five multidimensional stylized networks were generated for each possible configuration to be used as the true networks for evaluating link inference performance. Each of these was subsampled by subtracting integer weights from the edges, with the proportion removed varying from ten to fifty percent. The resulting five partial observations with edges missing completely at random (MCAR) [126, 166, 169] were then passed to the link inference algorithms, which were tasked with recovering the edge weights removed from the first layer of the first time period. For the algorithms that could only work on single-layer networks, only the first layer of the observed network was provided. In the experiments on dynamic multiplex networks, the algorithms designed for multiplex networks were provided each layer summed across all time slices, and the algorithms designed for dynamic networks were given the sum of all network layers for each time period. This allowed the algorithms to operate as they were originally intended.

3.2.5 Inferring Missing Links in Covert Networks

Data Sets

The link inference algorithms were also compared across three multiplex networks and three networks that were both multiplex and dynamic. These data sets are summarized in Tables 3.1 and 3.2. The first of the multiplex networks, the Belo Horizonte data set, comprises seven layers of connections between 77 criminals [189]. Derived from Brazilian police files, each layer represents cooperation between criminals on one of seven crime types: arms trafficking, kidnapping, weapon carrying, homicide, theft, drug trafficking, and ideological falsehood [189]. Notably, this was the only binary network considered in this work; all others contained edge weight information. The second network, formed from the Operazione Montagna data, instead covers 154 members of a Sicilian mafia, with one layer representing the number of in-person meetings surveilled by police and a second layer representing the number of phone calls captured by wiretaps [45, 76].

Additionally, a set of tweets discussing the 2020 stabbings in Nice, France, was used to evaluate the success of link inference on information subject to collection restrictions imposed by social media companies. Gathered using Twitter's streaming API [195] by tracking relevant hashtags and terms in October and November of 2020, this collection of more than 600,000 tweets [25] was filtered to find potential anti-France discussion using more than one hundred manually labeled hashtags. All available tweets authored by the accounts using these anti-France hashtags were then collected for the same time period as the original data set using the Twitter Academic Research API [194]. Though not all the tweets were still available, this second set provided a more complete record of these users' messages. This collection therefore produced two sets of tweets: the observed sample of tweets and the more comprehensive user timelines. In contrast to the randomized sub-sampling typically used to evaluate link inference methods, this pair of data sets presented an opportunity to test such algorithms on a real observed network and its more complete counterpart.

First, two-layer multiplex networks were formed separately for the observed sample and for the user timelines, with one layer representing the communication (i.e., the mentions, retweets,

Network	Nodes	Layers	Directed	Weighted	Density
Belo Horizonte	77 criminals	Drug trafficking co-offenses	No	No	0.0145
		Theft co-offenses	No	No	0.0104
		Weapon carrying co-offenses	No	No	0.0060
		Homicide co-offenses	No	No	0.0053
		Kidnapping co-offenses	No	No	0.0022
		Arms trafficking co-offenses	No	No	0.0005
		Ideological falsehood co-offenses	No	No	0.0003
Operazione Montagna	154 mafiosi	In-person meetings	No	Yes	0.0217
		Wiretapped phone calls	No	Yes	0.0105
French terror timelines	1000 Twitter users	Retweets + quotes + mentions + replies	Yes	Yes	0.0071
French terror sample	1000 Twitter users	Retweets + quotes + mentions + replies	Yes	Yes	0.0025
		Common hashtag usage	No	Yes	0.9552

Table 3.1: Summary of the real multiplex networks used in this study.

quotes, and replies) between the one thousand most active anti-France users and the other layer representing the number of times two people used the same hashtag in a tweet. Second, dynamic multiplex networks were also formed for the observed sample and for the user timelines. These networks were designed to test the algorithms' abilities to distinguish between interaction types, so the first layer was formed to represent direct communication between users, namely replies and mentions. The second layer was instead formed from retweets and quotes, since those are less direct forms of interaction. Given that these two layers both represented interactions between users, no normalization was applied. Finally, the temporal dimension of these data was represented by splitting the layers into four non-overlapping one-week time periods. Because the collected data spanned slightly longer than four weeks, some of the tweets used to create the multiplex networks were not considered when creating the dynamic multiplex networks.

A second dynamic and multiplex data set was created from the Violent Non-State Actor Rivalry (ViNSAR) project, which maps dyadic rivalries between 207 terrorist organizations in Africa and Asia from 1990 to 2015 [51]. Four layers were formed from these data based on the type of rivalry between two groups: violent rivalries that lead to deaths, other rivalries with reported violence, reported threats between groups, and denunciations of another group. These layers were separated into five non-overlapping five-year time periods representing changes in intergroup relations from 1990 through 2014. The weight of each edge therefore represented the number of years (up to a maximum of five) during which the given type of rivalry was present between two groups.

The last dynamic multiplex network was created based on messages posted from 2004 to 2010 to the Islamic Network, a former English-language forum for Muslims [197]. From this

Network	Nodes	Periods	Layers	Directed	Weighted	Density Min.	Density Mean	Density Max.
Dynamic French terror timelines	1000 Twitter users	4 weeks	Mentions + replies	Yes	Yes	0.0021	0.0027	0.0033
Dynamic French terror sample	1000 Twitter users	4 weeks	Mentions + replies	Yes	Yes	0.0001	0.0008	0.0014
			Retweets + quotes	Yes	Yes	0.0001	0.0007	0.0013
ViNSAR	207 terrorist groups	5 half-decades	Deadly violence	Yes	Yes	0.0006	0.0011	0.0014
			Other violence	Yes	Yes	0.0010	0.0012	0.0017
			Threats	Yes	Yes	0.0003	0.0004	0.0005
			Denunciations	Yes	Yes	0.0005	0.0009	0.0011
Islamic Network	529 forum users	27 quarters	Afghanistan	Yes	Yes	0	0.0005	0.0034
			Chechnya	Yes	Yes	0	0.0002	0.0019
			Guantanamo	Yes	Yes	0	0.0004	0.0035
			Israel-Palestine	Yes	Yes	0	0.0016	0.0092
			Kashmir	Yes	Yes	0	0.0002	0.0016

Table 3.2: Summary of the real dynamic multiplex networks used in this study.

forum, sets of keywords were used to identify the posts discussing five topics that can spark controversy on such fora: the war in Afghanistan, treatment in Chechnya, treatment in Guantanamo Bay, the conflict over Kashmir, and the Israel-Palestine conflict. These keywords are presented in Appendix C. Isolating these posts and their corresponding comments yielded a set of 6,064 messages from 529 users. Communication networks between these users were formed for each of the five topics by forming ties from a comment's author to the authors of the original post and any preceding comments, an approach also taken by Phillips et al. [158]. Finally, these different topic layers were divided into twenty-seven non-overlapping quarterly (i.e., three-month) time periods to add a dynamic component to the multidimensional network.

Multiplex Experiments

Given that the Belo Horizonte data were derived from police investigations, the corresponding link inference experiments sought to replicate the spotlight effect known to impact police data [41] by selecting specific individuals to highlight in the network. In each of several trials, a certain proportion of the seventy-seven criminals were chosen uniformly at random, and only their connections were retained in all seven network layers. This step simulated the process of arresting specific individuals and only discovering their co-offenders. From there, a small amount

of observation error was added to the data by removing ten percent of the remaining connections in each layer uniformly at random. This produced an observed network with edges missing at random (MAR) [126, 166, 169], since the spotlit individuals would have been known to law enforcement when they were arrested. However, only the observed network was subsequently passed to each of the link inference algorithms, which were tasked with identifying the missing connections in the densest layer, namely the one on drug trafficking. Since these layers varied drastically in density, the multidimensional link inference algorithms were given the option of normalizing each layer by its maximum link weight during the five-fold cross-validation process used to optimize the algorithm parameters. Five trials were run for each of various proportions of the nodes, ranging from 50% to 90%.

For the Operazione Montagna network, a similar procedure of sampling the mafia members was used. However, rather than sampling the individuals uniformly at random, they were sampled proportional to the square root of their total degree centrality across both layers. This process reflected law enforcement's limited surveillance resources and the need to prioritize more evident or important individuals. Moreover, since the second layer of the Operazione Montagna network was derived from wiretaps, the additional ten-percent observation error was only applied to the first layer derived from in-person surveillance. As with the Belo Horizonte data, five trials were conducted on observed networks with edges missing at random (MAR) generated using various proportions of the nodes as input. The link inference algorithms were tasked with recovering the missing connections in the in-person surveillance layer, and cross-validation was used to determine whether the multidimensional algorithms should normalize the network layers before running.

Since the French terror data were collected in two steps, first as a streamed sample and then as user timelines, they presented a unique opportunity to test whether the link inference algorithms could recover connections missing not at random (MNAR) [126, 166, 169] from the streamed sample. Consequently, only one experiment was run using the multiplex French terror network. Specifically, the network formed from the streamed sample was provided to the link inference algorithms, and their performance at recovering the communication layer (i.e., mentions, replies, retweets, and quotes) was measured against the connections present in the more complete timeline data. This was a particularly difficult task, since the communication layer in the sample network contained only 8.8% of the edges present in the timeline network. Since the two layers in this sample network differed significantly in connection type, the multilayer algorithms were allowed to normalize the layers by their maximum edge weight when optimizing their parameters via cross-validation.

Dynamic Multiplex Experiments

Similarly, one trial was run on the dynamic and multiplex version of the French terror data, again using the streamed sample network to infer the additional mentions and replies (not retweets or quotes) present in the last time period of the user timeline network. Unlike the experiment on the multiplex version of the data, this experiment therefore evaluated whether the link inference algorithms were able to recover different interaction types on the same platform. This was an even more difficult task, since the last time period of mentions and replies in the sample network contained 4.2% of the edges in the corresponding layer of the timeline network.

Since the layers and periods in these network data all represented similar connection types, no normalization was used. It should also be noted that some of the benchmark algorithms could handle networks that were only multiplex or dynamic, but not both. Since NCP4 was designed for dynamic networks [219], it was executed on dynamic multiplex networks by first summing the network across its layers but not its time periods. Since MLRW was designed for multiplex networks [141], it was instead run by summing the network across the periods but not the layers.

More neutral experiments were instead run on the ViNSAR terrorism network. Specifically, multiple experiments were conducted by removing edge weights from each layer and time period uniformly at random, thereby simulating edges missing completely at random (MCAR) [126, 166, 169]. As described in Chapter 1, the ViNSAR data are likely to over-represent the connections of certain groups. However, the extent of this bias in this specific network is not clear, nor is an appropriate sampling methodology given the large number of factors influencing such potential bias [64, 65, 206]. Furthermore, using uniformly random sampling on this network allowed for comparison against the other real networks in order to help understand the impact of the sampling process on link inference performance. This random sampling procedure was therefore carried out for several proportions of the ViNSAR data ranging from 50% to 90%, and five trials were carried out for each proportion. Each sampled network was given to the various link inference algorithms for determining whether they could identify the missing deadly rivalries in the last time period of the data set.

Finally, the Islamic Network data were used to determine whether the link inference algorithms could infer communication about a different topic. For each of the five topics, an experiment was conducted for which only the connections resulting from the other four topics were provided to the link inference algorithms. Each algorithm's ability to infer the connections corresponding to the fifth topic (regardless of time period) was then evaluated. This experiment therefore tested whether link inference could recover out-of-topic information if an analyst had not collected posts on that topic from the Islamic Network online forum.

3.2.6 Identifying Simulated Spurious Links

After comparing the performance of the aforementioned link inference algorithms for their intended task, simulation trials on stylized networks were used to instead evaluate their abilities to identify spurious network links. The same multidimensional stylized networks created for the link inference trials were used for this task, meaning that five true networks were used for each configuration of parameters. Rather than removing ties, a certain proportion of the total edge weight was added by incrementing (by one) the weights of edges chosen uniformly at random with replacement. The resulting augmented networks were passed to the link inference algorithms, which were tasked with recovering the first layer of the first time period. As with the link inference trials, single-layer algorithms were provided with only the first layer of the augmented network, whereas those designed for multiplex networks were given each layer summed across all time slices and those designed for dynamic networks were given the sum of all layers for each time period. Performance for identifying spurious links was measured analogously to the simulated link inference experiments, except that the likelihood scores computed by the link inference algorithms were negated to reverse their ordering. Furthermore, the testing sets comprised the edges that had been added to the observed networks rather than edges that had been removed.

3.3 Results

3.3.1 Simulated Missing Link Inference

Multiplex Stylized Networks

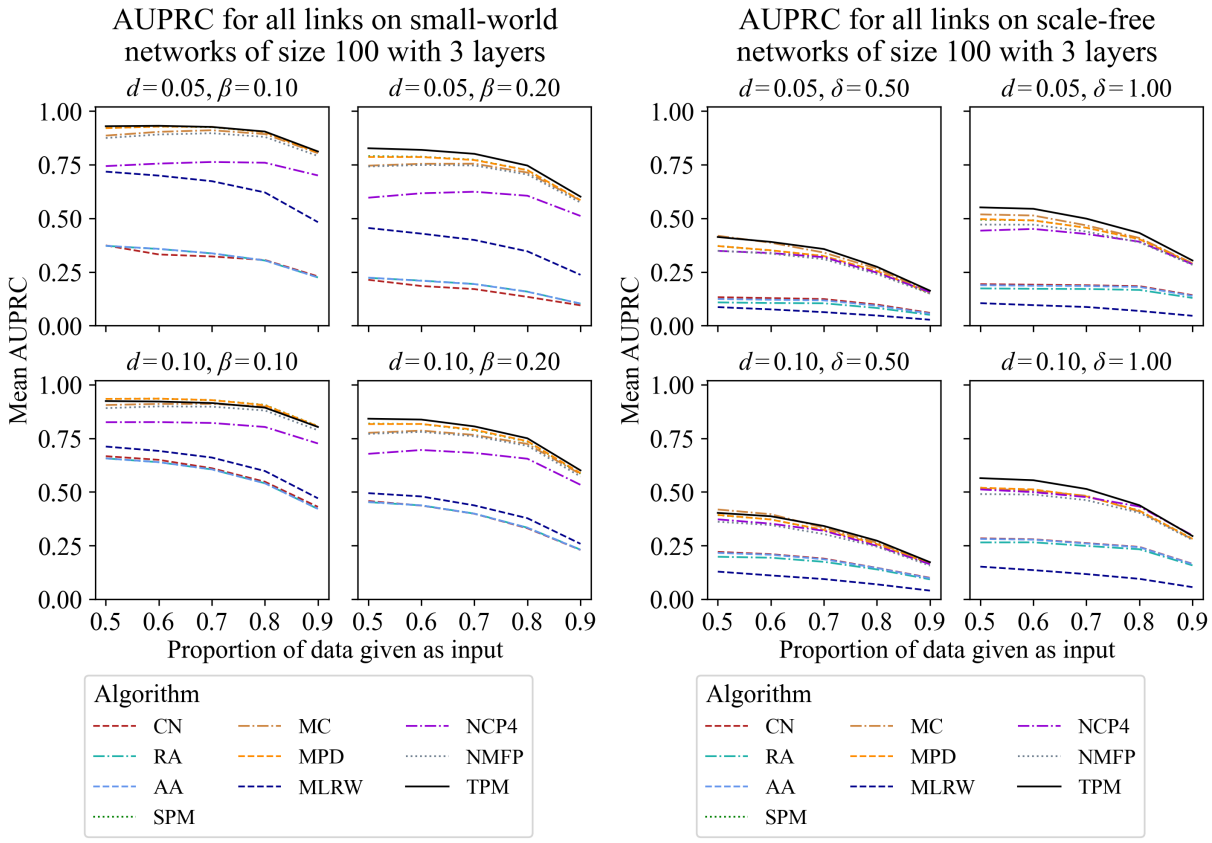
Figure 3.1a shows that TPM generally outperformed the AUPRC of other link inference algorithms on small-world networks of one hundred nodes, with the plot also demonstrating that it became more difficult to identify true positives as fewer and fewer edges were missing from the input data. In contrast, the AUROC results reported in Appendix D increase as a greater proportion of the edges are provided as input to the link inference algorithms. This behavior matches previous results showing that precision increases while AUROC decreases as more edges are missing from a network [152]. Furthermore, comparing the top subplots to the bottom ones shows that increasing the density of the small-world networks notably increased the link inference performance of common neighbors (CN) and its related algorithms, Adamic-Adar (AA) and resource allocation (RA). Meanwhile, comparing the left subplots to the right ones also reveals that increasing β, the probability of "long-distance" edges cutting across the small-world network, tended to decrease link inference performance. Finally, comparing Figure 3.1a to the results for fifty nodes shown in Appendix D demonstrates that increasing the network size from fifty to one hundred nodes only meaningfully increased the performance of CN, AA, and RA.

Figure 3.1b shows that link inference was more difficult on scale-free networks than on small-world networks, with AUPRC often below 0.5. In most cases, TPM continued to outperform the AUPRC of other algorithms, except when performance was calculated solely on edges that were completely absent from the input network (as shown in Appendix D). Unlike for the small-world networks, comparing the top and bottom subplots shows no notable difference in performance when changing the network density. However, comparing the left and right subplots reveals that increasing δ, the accumulation of weights in the preferential attachment process, notably increased link inference performance. Furthermore, comparing Figure 3.1b to the results in Appendix D shows no significant change in performance between the scale-free networks with fifty nodes and those with one hundred nodes.

The link inference results on uniformly random networks, which can be found in Appendix D, showed the lowest performance of all three network styles. TPM again often outperformed the other algorithms in AUPRC, but there was little change in performance when changing network size or density.

Dynamic Multiplex Stylized Networks

Figure 3.2a shows each algorithm's performance on small-world networks with one hundred nodes, three time periods, and three layers per time period. The results largely match those for the multiplex small-world networks, with TPM usually achieving the best AUPRC and increasing network size or density improving the performance of CN and its related algorithms. Furthermore, increasing the probability of "long-distance" edges again tended to decrease link inference performance. And just as with the non-dynamic small-world networks, comparing Figure 3.2a to the results for fifty nodes shown in Appendix D demonstrates that increasing the number of nodes from fifty to one hundred only notably increased the performance of CN, AA, and RA.



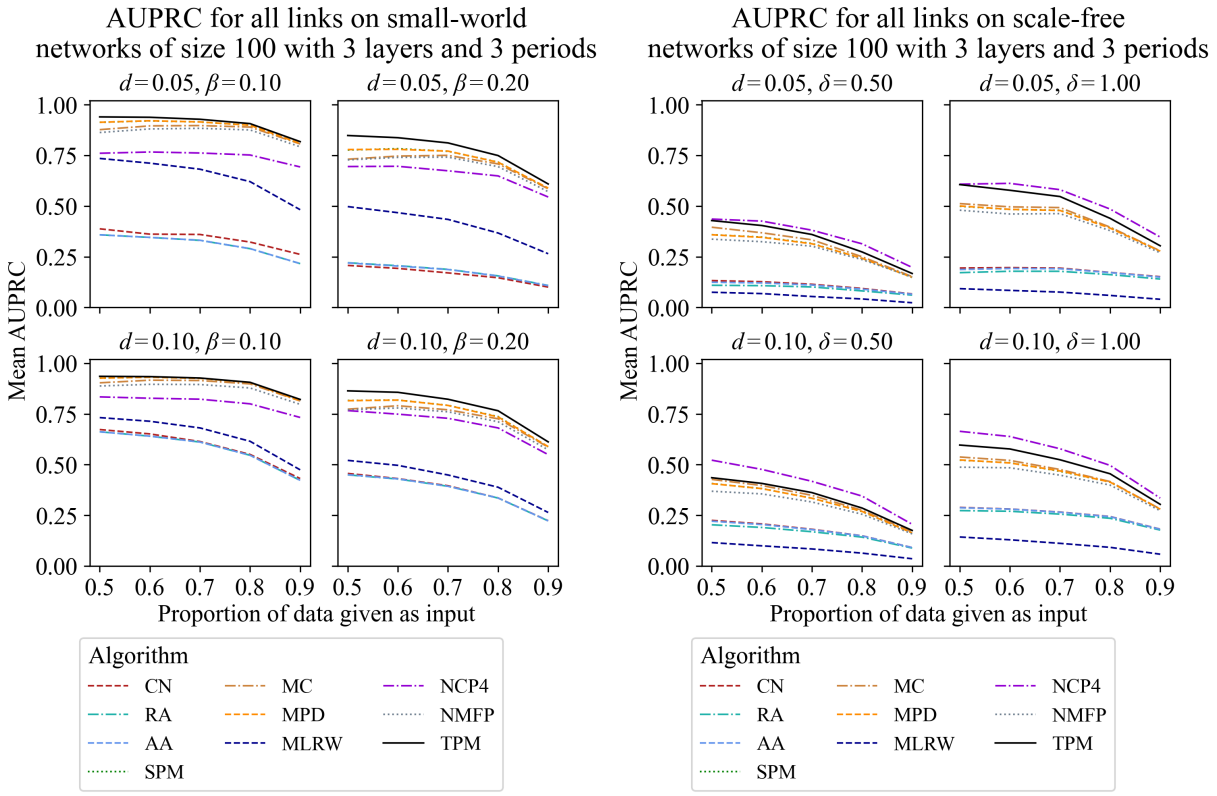
(a) Small-world networks

(b) Scale-free networks

Figure 3.1: Area under the precision-recall curve (AUPRC) for inferring missing links in multiplex stylized networks with three layers containing one hundred nodes. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density, β being the probability of "long-distance" edges in small-world networks, and δ dictating the accumulation of edge weights in scale-free networks.

Figure 3.2b again reveals that link inference was more challenging in scale-free networks. Like with the multiplex scale-free networks, increasing the accumulation of edge weight in the preferential attachment process tended to increase link inference performance. These results also show that NCP4 was usually better at identifying missing links in scale-free networks than TPM, and the corresponding figure in Appendix D shows that both NCP4 and matrix completion (MC) exceeded the AUPRC of TPM when calculated solely on edges that were completely absent from the input network. However, Appendix D also reveals that the increased performance of NCP4 came at the expense of runtimes that were sometimes more than five times as long as those of TPM. Comparing Figure 3.2b to the results in Appendix D also shows that increasing the number of nodes from fifty to one hundred did not significantly impact link inference performance, just as with the non-dynamic scale-free networks.

Finally, the results on uniformly random networks once more presented the lowest link in-



(a) Small-world networks

(b) Scale-free networks

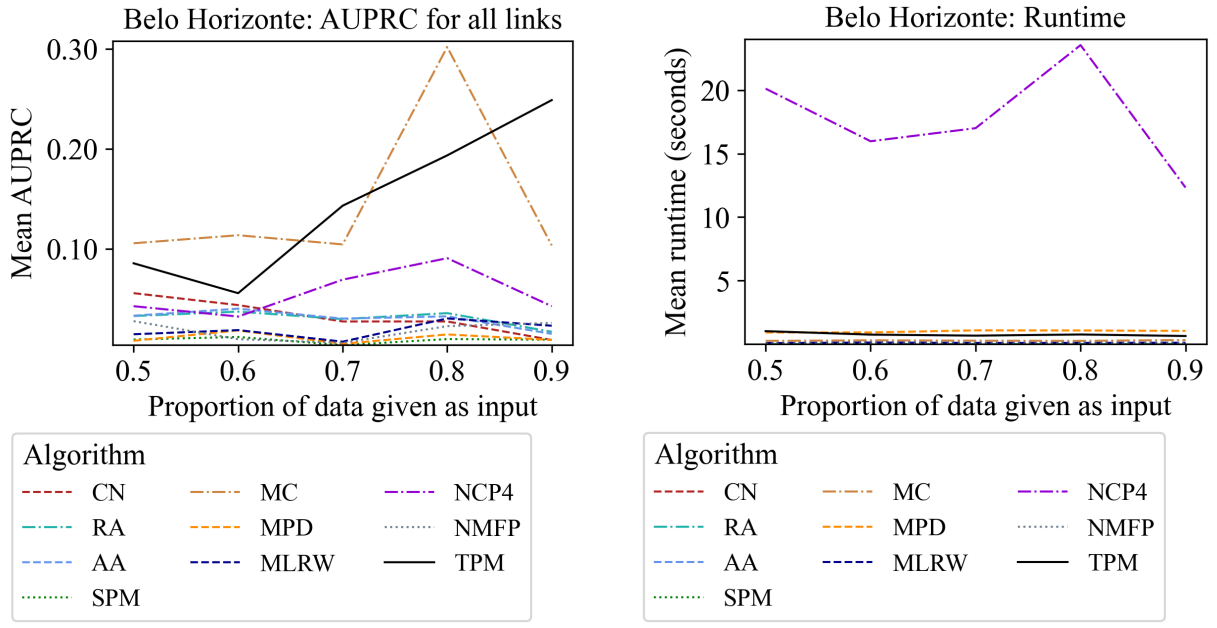
Figure 3.2: Area under the precision-recall curve (AUPRC) for inferring missing links in dynamic multiplex stylized networks with three time periods of three layers containing one hundred nodes. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density, β being the probability of "long-distance" edges in small-world networks, and δ dictating the accumulation of edge weights in scale-free networks.

ference performance with little discernible impact from network size or density. The results in Appendix D show that TPM and NCP4 often changed places as the best-performing algorithm in terms of AUPRC.

3.3.2 Covert Missing Link Inference

Multiplex Covert Networks

To evaluate TPM on real networks, its performance was measured on three multiplex networks in experiments intended to reflect realistic data collection scenarios. Figure 3.3 shows that the mean AUPRC achieved by TPM on the Belo Horizonte co-offenses network was only occasionally exceeded by that of the MC algorithm, with the precision results in Appendix E being rather similar except at the fifty- and sixty-percent marks. However, the performance of MC was much more variable due to its particularly high sensitivity to the λ parameter used in its principle



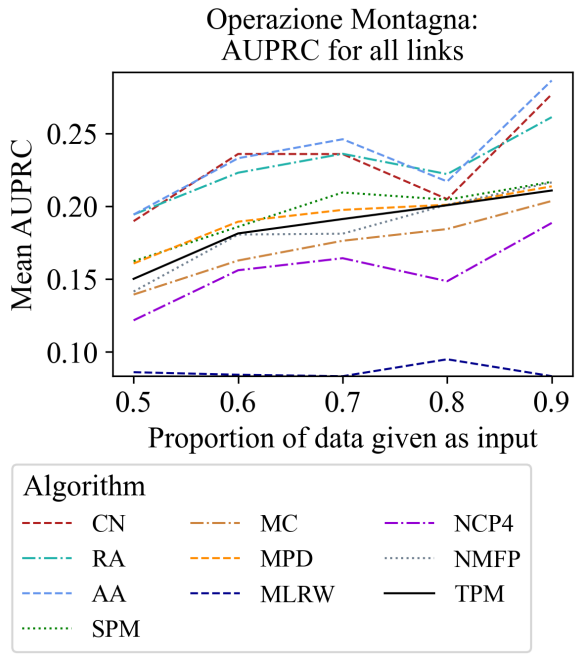
(a) Area under the precision-recall curve (AUPRC) for inferring missing links

(b) Runtime (not counting cross-validation)

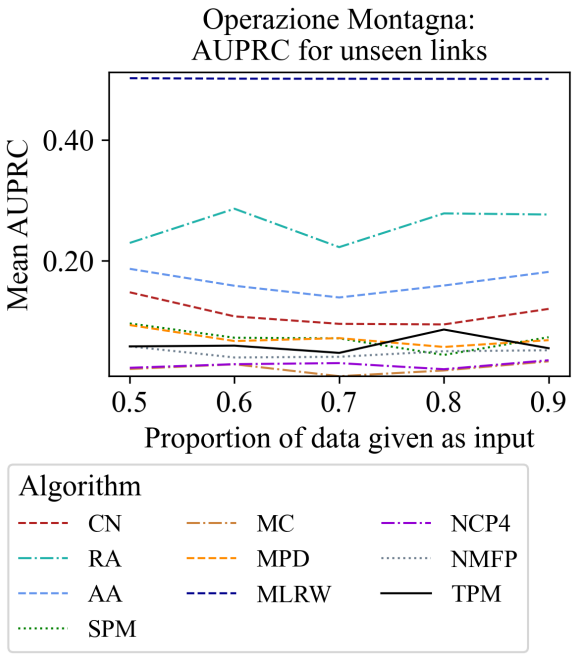
Figure 3.3: Performance when inferring drug trafficking co-offenses in the Belo Horizonte data [189]. The horizontal axes show the proportions of total edge weight provided as input.

component analysis, meaning that the best parameter value identified during cross-validation may not be the best for the full input network. In contrast, TPM required minimal optimization of parameters. Since the Belo Horizonte data constituted a binary network, there was no distinction between wholly unseen and partially weighted edges. In contrast, Figure 3.4 shows that the performance of TPM on the Operazione Montagna network ranked in the middle of the other link inference algorithms, with those based on CN (such as RA) performing significantly better. This also held true for the precision results, which are shown in Appendix E. Notably, both the Belo Horizonte and the Operazione Montagna networks saw link inference AUPRC increase or remain steady as fewer nodes' connections were removed from the networks, which comes in contrast to the decreasing AUPRC produced by the uniformly random link removal used in the experiments on stylized networks.

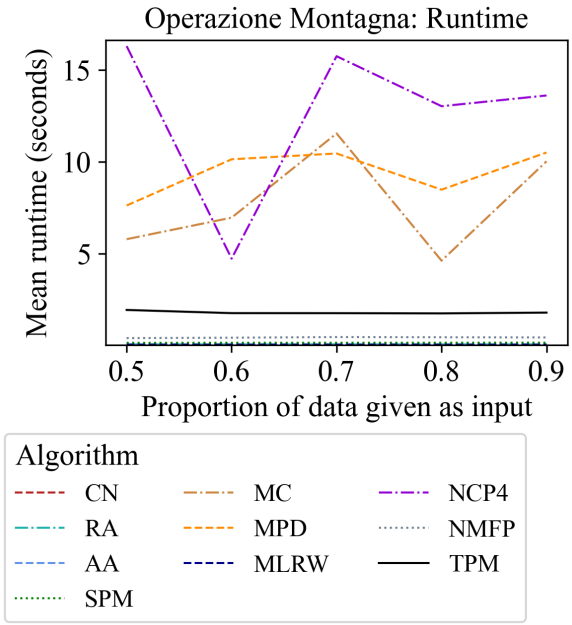
Figure 3.5 shows the results of inferring communication links in the multiplex network formed from the French terror tweets. TPM yielded a noticeable improvement in AUPRC over other algorithms, even when considering only wholly unseen links. However, the results in Appendix E show that TPM came second to multilayer random walks (MLRW) by a large margin in terms of precision. Overall, these results indicate the difficulty of this task, with AUPRC peaking at just under twenty-eight percent. This performance is not surprising, since the communication network formed from the original sample contained only 8.8% of the edge weights present in the network formed from the user timelines.



(a) Area under the precision-recall curve (AUPRC) for inferring *any* missing links

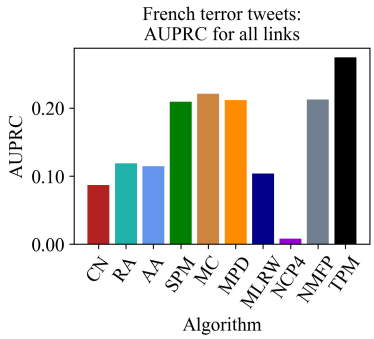


(b) Area under the precision-recall curve (AUPRC) for inferring *wholly unseen* links

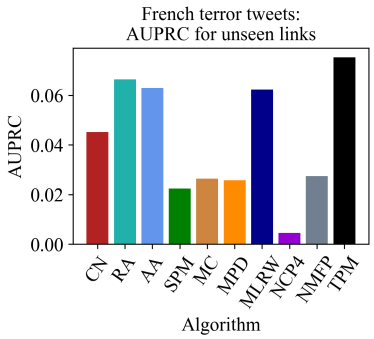


(c) Runtime (not counting cross-validation)

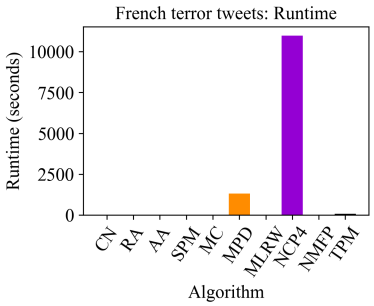
Figure 3.4: Performance when inferring in-person meetings between mafiosi in the Operazione Montagna data [45, 76]. The horizontal axes show the proportions of total edge weight provided as input.



(a) Area under the precision-recall curve (AUPRC) for inferring *any* missing links

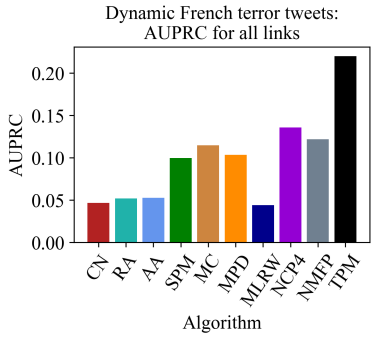


(b) Area under the precision-recall curve (AUPRC) for inferring *wholly unseen* links

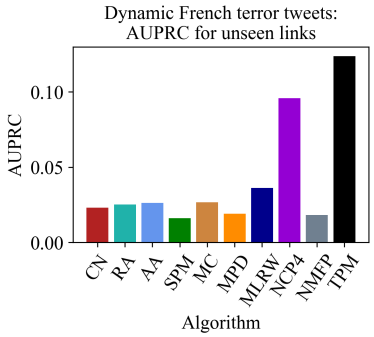


(c) Runtime (not counting cross-validation)

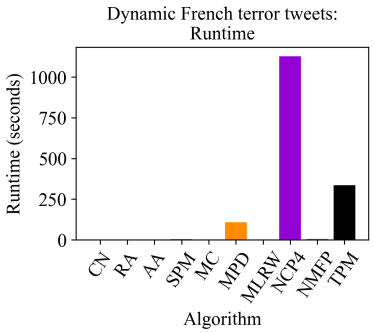
Figure 3.5: Performance when inferring communication between Twitter users discussing the 2020 French terror attacks.



(a) Area under the precision-recall curve (AUPRC) for inferring *any* missing links



(b) Area under the precision-recall curve (AUPRC) for inferring *wholly unseen* links



(c) Runtime (not counting cross-validation)

Figure 3.6: Performance when inferring reply and mention connections between Twitter users discussing the 2020 French terror attacks.

Dynamic Multiplex Covert Networks

The link inference algorithms were also tested on real networks that were both dynamic and multiplex. Figure 3.6 shows the link inference algorithms' performance when filling in the missing replies and mentions in the dynamic multiplex network formed from the French terror tweets. As also shown in Appendix E, TPM exceeded the performance of the other algorithms by all metrics except precision calculated only on wholly unseen links, for which NCP4 only slightly outperformed TPM. Nevertheless, the AUPRC never exceeded twenty-two percent, demonstrating again the difficulty of this task. Indeed, the last time period in the sample network contained only 4.2% of the mentions and replies in the last period of the timeline network, making this an extremely challenging link inference task.

In a second experiment on the ViNSAR network summarized in Figure 3.7, TPM achieved

performance rivaling the best link inference algorithms when considering all missing links but was not as successful at identifying wholly unseen links. This was broadly true when providing different proportions of the data as input to the link inference algorithms, and the precision reported in Appendix E generally decreased as fewer connections were missing from the input data. At times these precision results exceeded fifty percent, meaning this task was significantly easier than working with the French terror tweets.

Finally, Figure 3.8 shows that across all five concerning topics in the Islamic Network data, TPM either rivaled or exceeded the performance of all other algorithms except NCP4. In particular, the topics with the lowest communication density, which should therefore be the most difficult to infer, produced noticeable performance improvements when using TPM or NCP4. However, it should again be noted that NCP4 came at the expense of severely longer runtimes, as shown in Figure 3.8(c). Overall, the difficulty of this task varied greatly depending on the topic, with the Chechnya discussion yielding zero precision and the Israel-Palestine discussion yielding precision that could exceeded seventy percent (as reported in Appendix E).

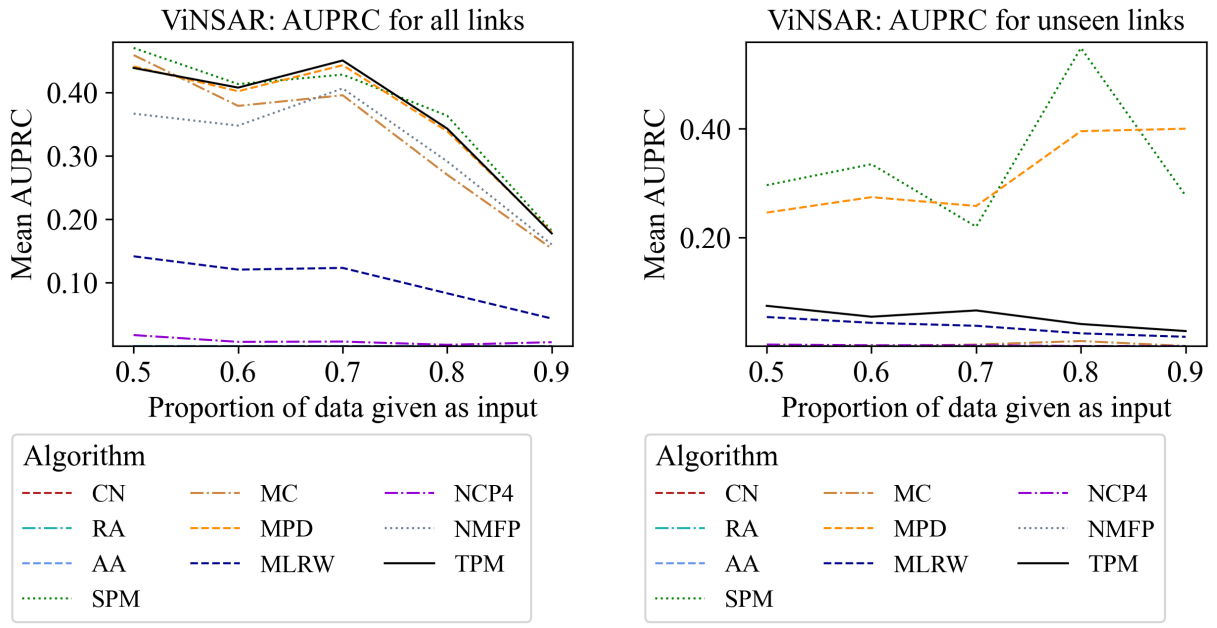
3.3.3 Simulated Spurious Link Identification

Multiplex Stylized Networks

Figure 3.9a shows that the link inference algorithms performed remarkably well at identifying spurious links on small-world networks of one hundred nodes, with TPM generally outperforming the AUPRC of other algorithms. As more spurious edges were introduced to the networks, most algorithms' AUPRC tended to increase, reflecting the behavior observed in the missing link inference experiments. This comes in contrast to the AUROC results reported in Appendix F, which remain roughly constant regardless of the number of spurious links in the small-world networks. As with the experiments on missing links, increasing network density increased the AUPRC of CN and its related algorithms, and increasing the probability of "long-distance" edges tended to decrease performance. As shown in Appendix F, situations with higher β and lower density saw the MC and MLRW algorithms outperform TPM in terms of precision. Finally, comparing Figure 3.9a to the results for fifty-node networks shown in Appendix F reveals that changing network size increased the AUPRC of CN, AA, and RA.

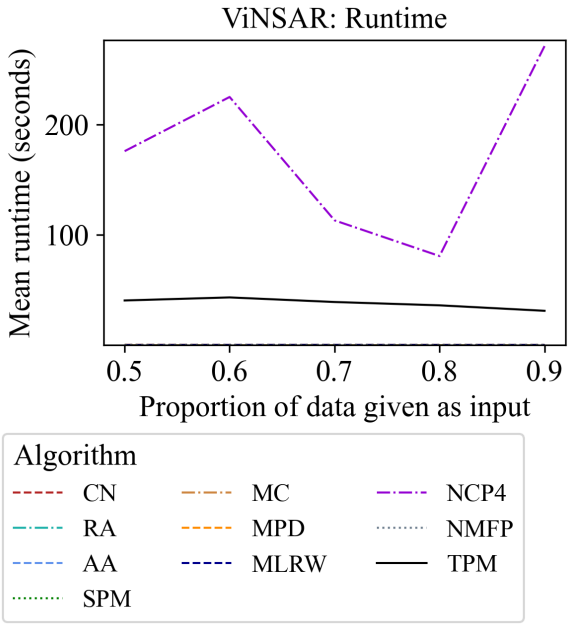
Figure 3.9b shows that identifying spurious links in scale-free networks was much more difficult. TPM often yielded a higher AUPRC but was occasionally outmatched by the MC algorithm. In contrast, the precision results in Appendix F show that MC and MLRW always outperformed TPM and the other algorithms. As with the small-world results, AUPRC tended to increase as more spurious edges were introduced into the scale-free networks, whereas AUROC remained roughly constant. Matching the missing link inference experiments, increasing the accumulation of edge weights in the preferential attachment process also tended to increase performance. Lastly, comparing Figure 3.9b to the fifty-node results in Appendix F shows that only the performance of NCP4 was impacted by the change in network size, with the larger networks yielding slightly lower NCP4 performance at identifying spurious links.

The results for identifying spurious links in uniformly random networks can be found in Appendix F. As with the missing link inference experiments, these networks yielded the lowest AUPRC, with MC and MLRW always outperforming TPM. Increasing network size or density



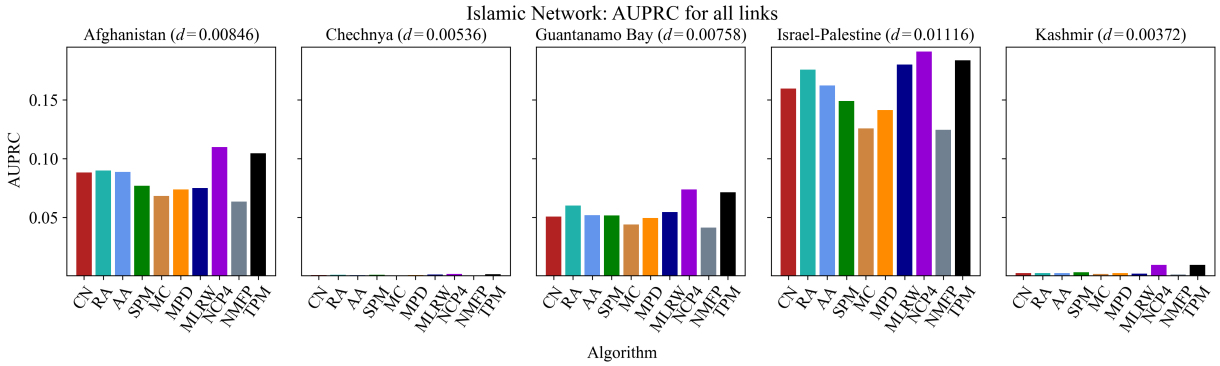
(a) Area under the precision-recall curve (AUPRC) for inferring *any* missing links

(b) Area under the precision-recall curve (AUPRC) for inferring *wholly unseen* links

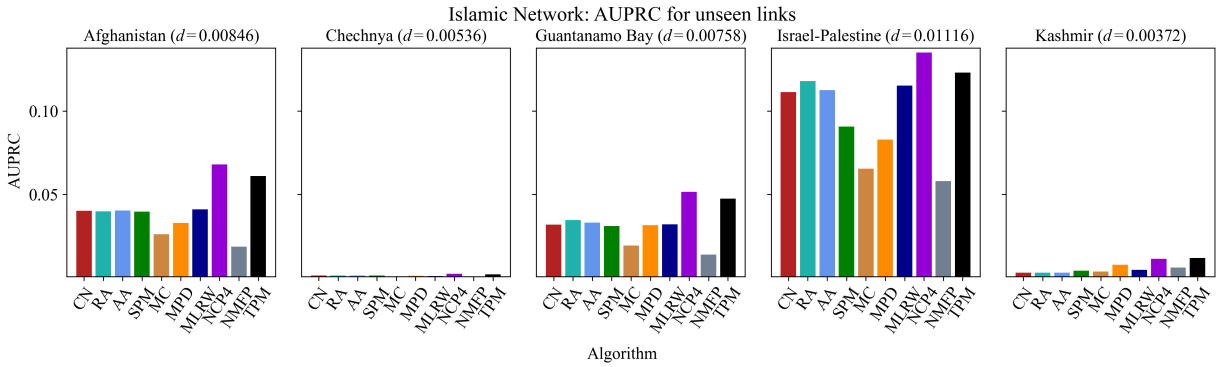


(c) Runtime (not counting cross-validation)

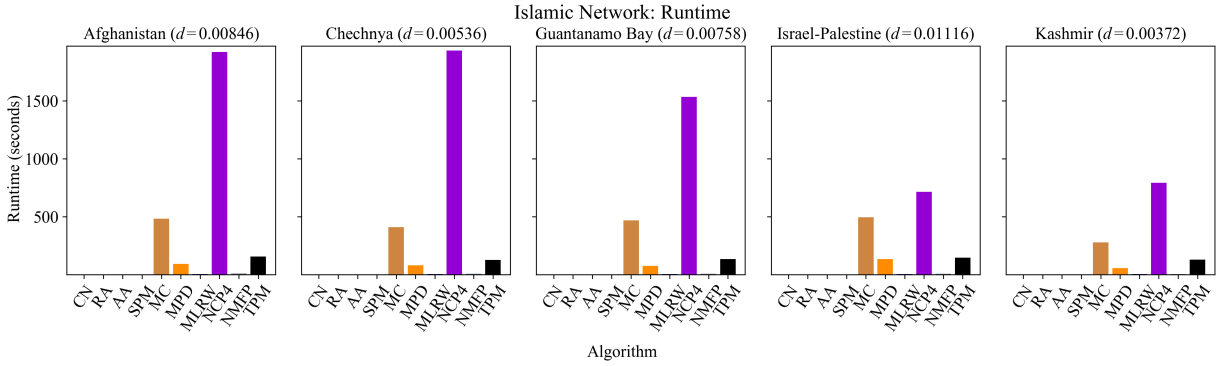
Figure 3.7: Performance when inferring deadly rivalries between terrorist groups in the ViNSAR data set [51]. The horizontal axes show the proportions of total edge weight provided as input.



(a) Area under the precision-recall curve (AUPRC) for inferring *any* missing links

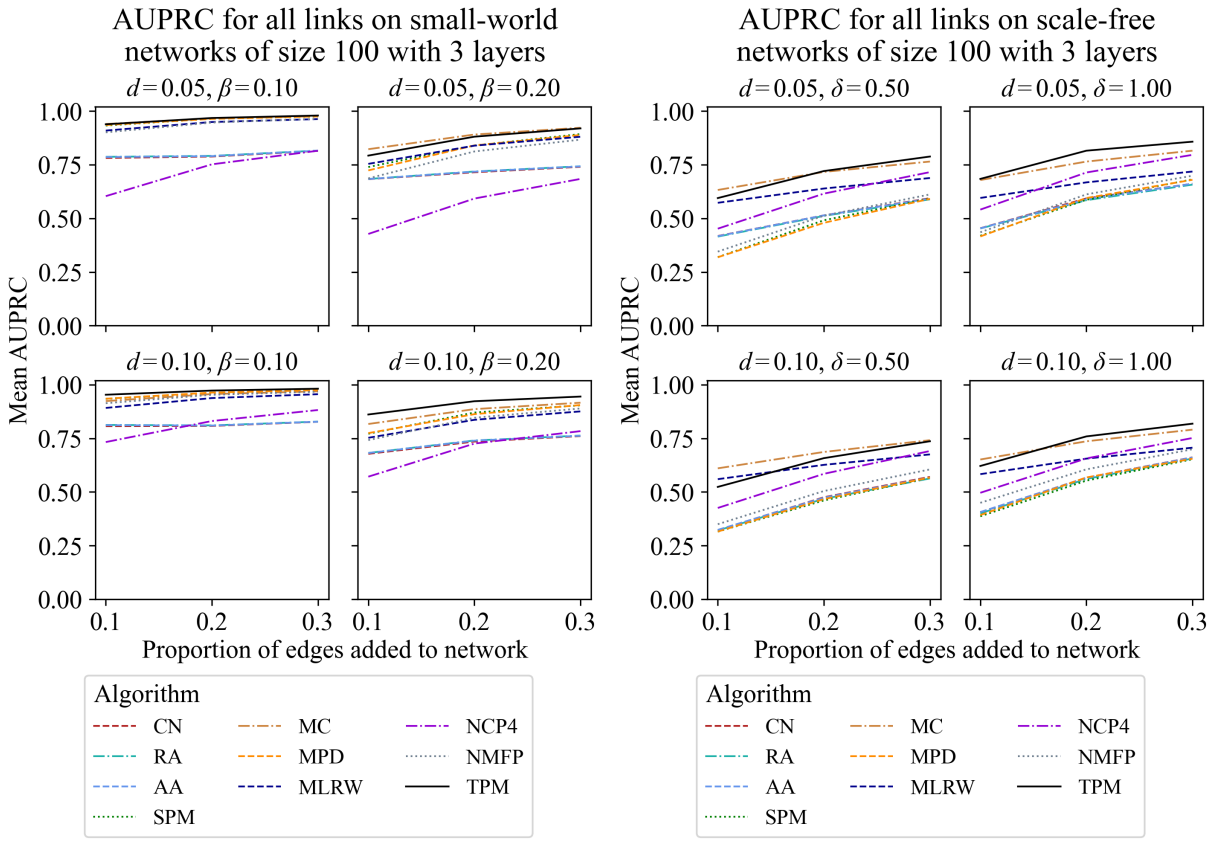


(b) Area under the precision-recall curve (AUPRC) for inferring *wholly unseen* links



(c) Runtime (not counting cross-validation)

Figure 3.8: Performance when inferring topic-specific communication in the Islamic Network online forum [197]. Each subplot shows the performance for inferring communication on that topic when provided with the communication related to the other four topics. Each subplot title also indicates the density d of the communication network for that topic.



(a) Small-world networks

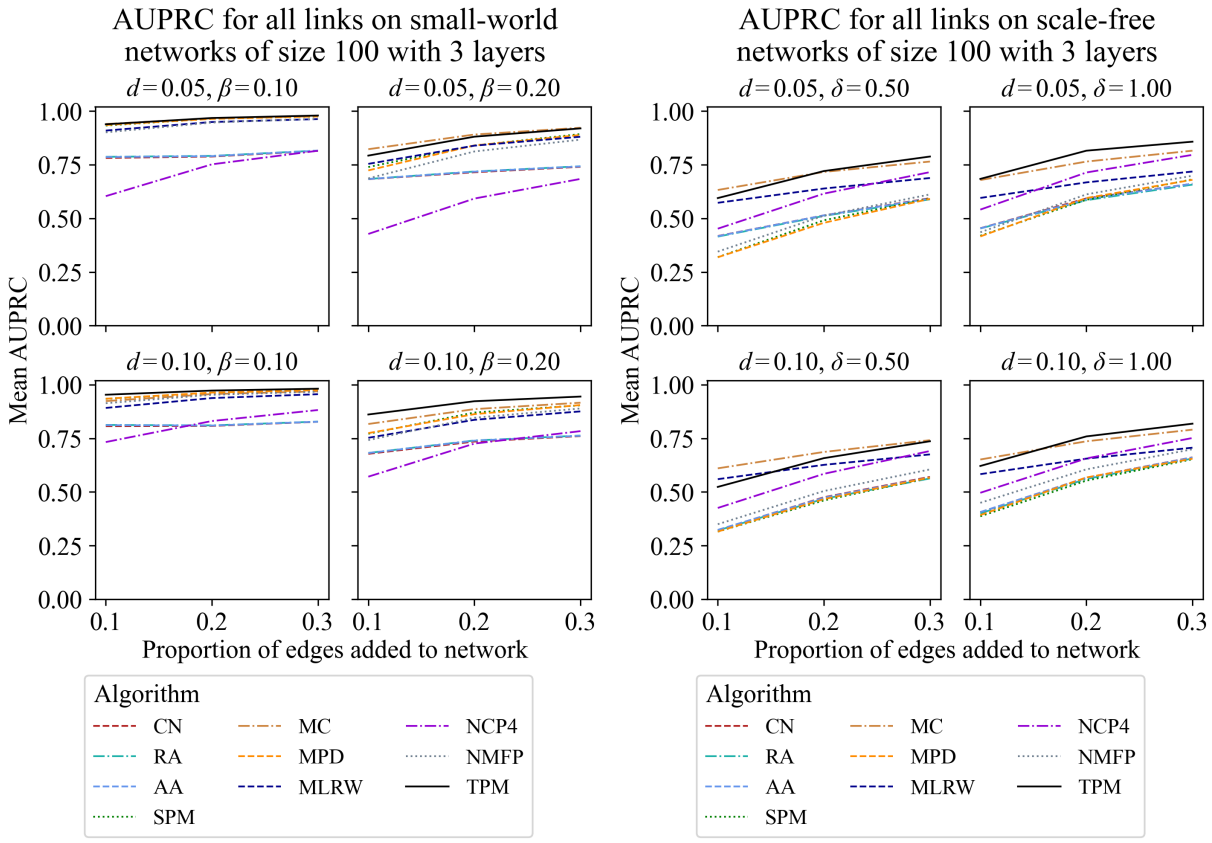
(b) Scale-free networks

Figure 3.9: Area under the precision-recall curve (AUPRC) for identifying spurious links in multiplex stylized networks with three layers containing one hundred nodes. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density, β being the probability of "long-distance" edges in small-world networks, and δ dictating the accumulation of edge weights in scale-free networks.

only slightly decreased the performance of TPM and NCP4, and increasing network density decreased the performance of CN, AA, and RA.

Dynamic Multiplex Stylized Networks

Figure 3.10a shows each link inference algorithm's performance when adapted to identifying spurious links in dynamic multiplex small-world networks. The results largely match those for multiplex small-world networks, with TPM usually yielding the highest AUPRC and performance increasing as more spurious edges were added to the network. Moreover, increasing network density increased the AUPRC of CN and its related algorithms, AA and RA, and increasing the probability of "long-distance" edges tended to decrease performance. Finally, comparing Figure 3.10a to the results for fifty-node networks included in Appendix F reveals that increasing



(a) Small-world networks

(b) Scale-free networks

Figure 3.10: Area under the precision-recall curve (AUPRC) for identifying spurious links in dynamic multiplex stylized networks with three layers containing one hundred nodes. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density, β being the probability of "long-distance" edges in small-world networks, and δ dictating the accumulation of edge weights in scale-free networks.

network size increased the AUPRC of CN, AA, and RA.

Figure 3.10b again demonstrates that identifying spurious links was more difficult in scale-free networks. Like with the multiplex scale-free networks, increasing the accumulation of edge weights tended to increase performance. Unlike in the experiments for inferring missing links, TPM usually matched or outperformed the other algorithms at identifying spurious links. Comparing Figure 3.10b to the fifty-node results in Appendix F reveals that increasing network size only notably decreased the AUPRC of NCP4.

Finally, the results on uniformly random networks in Appendix F again exhibit the lowest AUPRC for identifying spurious links. As with the multiplex networks, AUPRC decreased slightly for TPM and NCP4 when increasing network size or density, and increasing network density yielded decreases in the performance of CN, AA, and RA.

3.4 Discussion

3.4.1 Missing Links

With few exceptions, the experiments on multidimensional stylized networks showed TPM matching or exceeding the AUPRC of other link inference algorithms when considering all connections that were missing completely at random (MCAR). This was also the case for most of the experiments on real networks with edges missing at random (MAR), except for the multiplex Operazione Montagna mafia network. Though NCP4 at times exceeded the performance of TPM, it came at the expense of severely longer runtimes from increasing the dimensionality of the data before decomposing the network tensor. Since NCP4 generates its edge likelihood scores based on the hub and authority scores of nodes in the network, it should be particularly useful when nodes' connection behaviors are correlated to these factors but may not be as useful on other networks. As previous research has shown, different link inference algorithms work best on different kinds of networks [82].

Moreover, TPM was the best at recovering connections missing not at random (MNAR) from the streamed Twitter sample when compared to the corresponding user timelines, thereby providing remarkably realistic evidence of effectiveness when compared to the exclusively random sub-sampling used in most link inference studies. This makes TPM a good option for analysts working with incomplete networks of arbitrary dimensionality. Though prior research has shown that no link inference algorithm is best on all types of networks [82], TPM is a flexible starting point that accepts many kinds of data and can be used with little adjustment of parameters.

The trials on stylized networks also showed that small-world networks tended to be the easiest to infer, whereas uniformly random networks were the hardest. Increased randomness overall decreased link inference performance, with additional "long-distance" edges making inference more difficult in small-world networks. Furthermore, networks with more edge information, either from increased density or greater weights (in the case of scale-free networks), tended to make link inference easier. Comparing the experiments on real networks to those on stylized networks also suggests distinct influences on how link inference performance changes with the amount of missing connections. In the experiments relying on uniformly random sub-sampling to create "observed" networks for inference (including the trials on the ViNSAR terrorist rivalries network), true positives decreased as fewer ties were removed from the original network. In contrast, the experiments that removed connections based on selecting specific nodes, such as those using the Belo Horizonte co-offender network and the Operazione Montagna mafia network, saw increasing true positives as fewer nodes were selected for edge removal. This unfortunately means there is a greater likelihood of false positives when working with police data, since they may be collected from observing comparatively few individuals.

This work also highlights the need for link inference algorithms that function well on networks that are multidimensional, weighted, and directed. As the presented results have shown, this leads to important considerations for how to measure the performance of these algorithms, particularly whether to account only for ties that are completely unobserved in the input network or whether to also account for ties that have been observed with lower weight than in reality. Past research that considered only wholly unseen connections found that unweighted measures like resource allocation (RA) may still be best at inferring links in weighted networks due to the

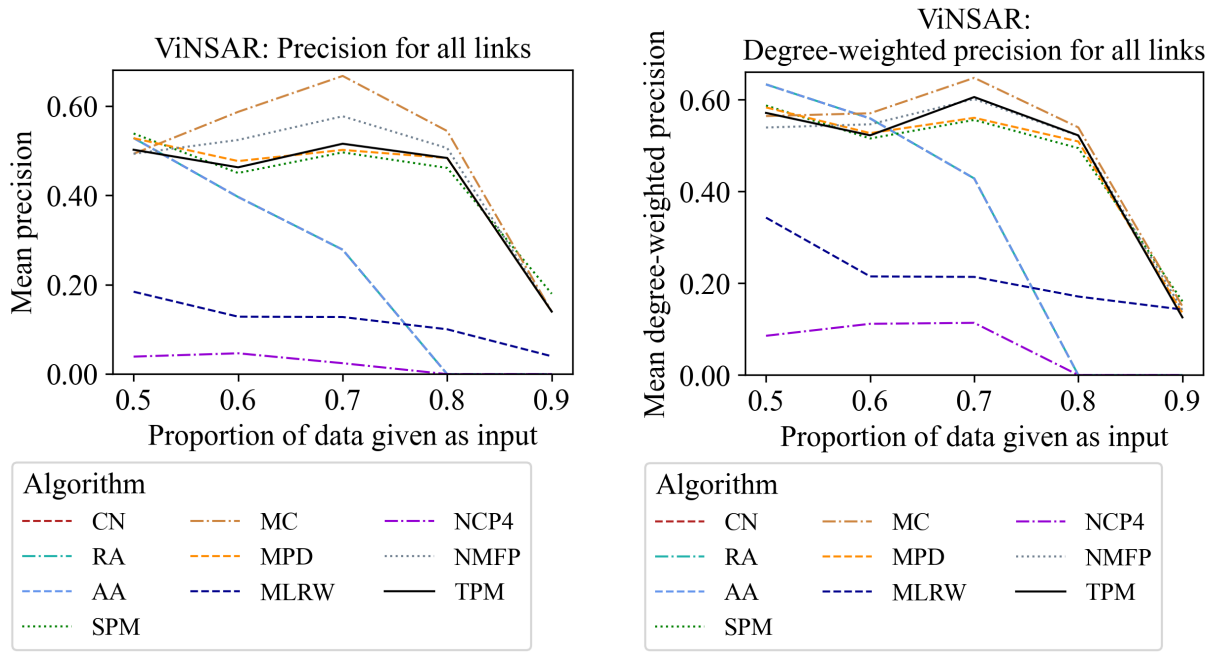
importance of weak ties [127], but the results presented here tell a more nuanced story, with the "best" algorithm depending on which kinds of missing connections are considered in evaluation. Future research on link inference should more carefully consider these problems, particularly given the weighted nature of the evermore-valuable networks formed from social media, mass transportation, and other systems [58]. When analyzing weighted networks using techniques such as centrality measures, inaccurate edge weights can degrade results just as much as completely unobserved connections.

Another important matter is the kinds of missing edges that link inference algorithms are more likely to identify. A social network analyst may be more interested in edges with particular characteristics, such as those in the core of a network or those on its periphery. Since different algorithms are known to work better than others depending on the type of data being studied [82], it is reasonable to assume that these algorithms may be better at inferring certain kinds of edges than others. Indeed, Figure 3.11 reveals changes in precision performance when weighting the inferred connections based on the degree centrality of the nodes they would connect. On the ViNSAR network of terrorist rivalries, the precision (i.e., true positive rate) of the link inference algorithms rose almost across the board when placing greater weight on "core" edges (i.e., ties that would connect higher-degree nodes). This is to be expected, since higher degrees indicate that more information is known about that node's connection behavior. However, Figure 3.11 also shows that certain algorithms' performance changed more than others'. Notably, the peak performance of MC was slightly lower compared to raw precision and was closer to the performance of TPM and other algorithms. On the other hand, MLRW exhibited significantly higher performance when measured as degree-weighted precision. This indicates that MLRW is much stronger at inferring core edges than peripheral ones, whereas MC may be better at inferring peripheral edges than other algorithms. More research should be done on this topic, but it is clear that many of these link inference algorithms are stronger at identifying edges in the core of a network and that this distinction varies from algorithm to algorithm.

3.4.2 Spurious Links

It may perhaps seem surprising that many of the link inference algorithms performed better at detecting spurious edges than at identifying missing links, but this finding matches the results presented by other studies using link inference algorithms to identify spurious edges [91, 161, 220]. These experiments show that this finding extends to multidimensional networks. Like in the experiments on inferring missing links, TPM was rarely outperformed by other algorithms. Given that TPM works natively on multidimensional networks, this research demonstrates the viability for removing spurious links from dynamic multiplex networks, a task that has thus far been neglected by prior studies.

In terms of the factors impacting performance at identifying spurious links, the results largely matched those for inferring missing links. Uniformly random networks were the hardest to work on, and small-world networks were the easiest. Increasing randomness by adding more "long-distance" edges to small-world networks decreased performance, and increasing density or amplifying edge weights tended to make identifying spurious links slightly easier. Few of the algorithms appeared to be noticeably impacted by changing the size of a network. In conjunction with the results from the experiments on missing links, these findings consolidate some intuitive



(a) Precision for inferring *all* missing links

(b) Degree-weighted precision

Figure 3.11: Precision when inferring deadly rivalries between terrorist groups in the ViNSAR data set [51]. The horizontal axes show the proportions of total edge weight provided as input. Degree-weighted precision is calculated by weighting each inferred edge by the minimum degree of the nodes that would be connected by the edge.

principles about the inference of stylized networks: greater randomness naturally makes them harder to infer (both for missing and spurious connections), and increased edge information makes them easier to infer.

3.5 Conclusion

3.5.1 Limitations

This work only tested one possible implementation of TPM on a limited number of data sets. In actuality, TPM provides a flexible framework for conducting multidimensional link inference that can be modified and adjusted using domain expertise. Rather than removing edges uniformly at random, the noise model used in the first perturbation step could instead be formulated to more closely reflect the processes causing edges to be missing in the specific network being studied, which could potentially improve the algorithm's performance on that data set. Additionally, the tensor provided to TPM could be constructed to represent domain knowledge, such as by including network layers that connect nodes sharing a common attribute. Since node attributes can significantly influence relationships (e.g., via homophily), providing this information to TPM could improve its performance. However, these possibilities were not explored in this work due

to time constraints and the need to first demonstrate the overall effectiveness of TPM.

Moreover, prior work has shown that different link inference algorithms function better on different types of networks [82], so the strengths of TPM identified in this research may not extend to other data. It is very likely that combining TPM with other link inference algorithms in an ensemble method would allow for better performance in a greater variety of contexts, as has indeed been shown with other algorithms [82]. Additionally, more research should be done on how the observation process impacts link inference performance, which has only recently garnered more research attention [95]. This work tested a few sampling strategies based on spotlighting high-degree nodes or removing edges uniformly at random, but this comparison was made across different data sets due to time constraints on testing different network configurations and observation processes. Evaluating the identification of spurious links on real multidimensional networks would also improve this line of research, and the addition of more algorithms (both for inferring missing links and identifying spurious edges) would strengthen this work.

In particular, comparing TPM to deep learning algorithms would be valuable to researchers developing future link inference methods. Several link prediction algorithms for dynamic multiplex networks already benefit from using deep learning but also suffer from the ensuing drawbacks. Xue et al. [212] and Cai et al. [36] applied attention mechanisms to recurrent neural networks and graph neural networks, respectively, whereas Ozcan and Oguducu [150] used nonlinear autoregressive neural networks. These models can yield powerful performance but are also expensive to train and require parameters to be tuned for each data set. Additionally, they must be trained for each specific task, such as predicting the next time step, and are not so easily adapted to networks of different dimensionalities, such as those that are only dynamic or only multiplex. However, comparing the performance of these algorithms to that of TPM could identify the strengths and weaknesses of each approach.

3.5.2 Contributions

This chapter presents an effective link inference method for multidimensional networks and contributes to the evaluation of link inference algorithms at large. The proposed algorithm, the tensor perturbation method (TPM), extends a matrix-based algorithm [204] to multidimensional tensors by repeatedly perturbing, compressing, and reconstructing the input network. Unlike other link inference methods developed for multidimensional networks, TPM can be used on networks that are multiplex, dynamic, or both multiplex and dynamic.

Simulated experiments on stylized networks showed that it generally matched or outperformed nine other link inference algorithms at filling in missing connections. Additional experiments on six real networks evaluated the performance of TPM in situations mimicking the ways in which covert network data might be collected and analyzed, such as by collecting conversations on one medium or topic and wanting to infer other types of communication. Some of these experiments relied on two social media data sets that together constituted a real "observed" network and its more complete counterpart, thereby providing a realistic opportunity for measuring link inference performance not available to most link inference studies. In all but one of these experiments on real networks suffering from three types of missingness [126, 166, 169], TPM generally matched or outperformed the other algorithms, except for a more complex algorithm [219] with runtimes multiple times as long as those of TPM.

Moreover, this research demonstrates the effective identification of spurious links in multidimensional networks. Simulated experiments on stylized networks showed that, as has been demonstrated for standard networks [91, 161, 220], link inference algorithms can be used to identify spurious connections in dynamic multiplex networks by examining the low-scoring links. These simulations again showed that TPM generally matched or outperformed the other link inference algorithms at detecting spurious ties.

These evaluation experiments also corroborate certain principles on how network topology and the pattern of missing ties can impact link inference performance. Greater randomness in the actors' connections tended to make link inference harder, whereas greater edge information (either as higher density or larger edge weights) tended to make link inference easier. Meanwhile, the differences between the experiments on stylized networks and those on real networks suggest that, as fewer and fewer data are collected, spotlighting particular nodes during collection makes link inference more difficult than introducing uniformly random error. This unfortunately means that police data in particular may be more problematic candidates for link inference, since they may be collected by spotlighting only a few individuals in a network. More broadly, these results reveal that the pattern of missing edges impacts not just the overall performance of a link inference algorithm [95] but also how that performance changes with the amount of missing data.

Beyond the significance of considering missing connections that are not uniformly random, this research also embraces the importance of studying link inference on networks that are weighted and directed, something that is often overlooked in link inference research [127]. Since the collection of such network data can often result in observing edge weights that are too low, this study presents separate performance metrics when calculated solely on connections that are wholly missing from the input networks and when also considering edges that were present in the input network but had inferior weights than in the true network. Often, the best algorithm for inferring wholly unseen edges was not the best algorithm overall. In addition to presenting a link inference algorithm for multidimensional algorithms, this work therefore raises important considerations for evaluating link inference methods on weighted, directed networks.

Chapter 4

Bounding Network Metrics[1]

4.1 Introduction

4.1.1 Motivation

When strategies for improving data quality have been exhausted, a remaining approach for coping with imperfect information is to quantify the resulting level of uncertainty in one's conclusions. For analyses that rely on network metrics like node centralities, confidence intervals have the potential to gauge the range of possible values in the unseen true network. Though some researchers have explored different methods of calculating confidence intervals for network metrics, many of these efforts [3, 68, 136, 187] have been dedicated to evaluating the statistical significance of observed metrics with respect to a null model. Only a few studies [47, 78, 162] have instead considered the problem of using an imperfect observed network to put bounds on the value of a network metric in the underlying complete network. However, most of these works focused on modeling the distributions of specific metrics and were not extended to weighted networks [47, 162].

The need for more general confidence interval methods instead points to other research using network models to bootstrap intervals for populations of networks [86, 123]. This approach calls for fitting a model to the observed network, using the model to generate a set of replicate networks that are similar to the observed data, calculating the desired metric on each of those replicates, and deriving a confidence interval from those metric values. Though several kinds of network models could be used for this purpose [87, 123, 170], Levin and Levina [123] showed that the random dot product graph (RDPG) yields intervals with good coverage for quantities like triangle density when looking to bound a metric for a population of networks. This methodology could be combined with techniques for fitting network models to imperfect network data in order to explicitly account for the possibility of missing edges and instead bound a metric for an unseen underlying network.

Moreover, the bootstrapping approach of repeatedly generating possible observations closely resembles the operation of link inference algorithms that repeatedly perturb a network's adja-

[1]Parts of this chapter were published in *Social, Cultural, and Behavioral Modeling: 18th International Conference, SBP-BRiMS 2025* [16]. The article is available on request.

cency matrix and average the results [129, 204]. Rather than averaging them together, each modified adjacency matrix can instead be interpreted as a replicate network for bootstrapping confidence intervals. Given these diverse possibilities for generating confidence intervals for network measures, a comprehensive comparison of these methods would be valuable for analysts wishing to quantify the uncertainty in their work.

4.1.2 Objectives

This chapter evaluates the effectiveness of different methods for bootstrapping networks when using them to construct confidence intervals for metrics calculated on incomplete networks. The task is therefore to use an observed incomplete network to compute probabilistic bounds for the value of a metric if it were calculated on the unseen complete network. More specifically, for a certain observation process $\mathcal{O}$, network metric m, and significance level $\alpha \in [0, 1]$, the goal is to use a set of observations $\boldsymbol{N} = \{\boldsymbol{N}_i | \boldsymbol{N}_i \sim \mathcal{O}(\boldsymbol{T})\}$ to compute bounds (l, u) such that $P(l \leq m(\boldsymbol{T}) \leq u) \geq \alpha$ for all possible true networks $\boldsymbol{T}$.

Since analysts often only have one observed network (i.e., $|\boldsymbol{N}| = 1$), the strategy of bootstrapping makes use of the observed network to generate a set of possible observations, hereafter referred to as replicate networks, that are then used to compute a confidence interval. As mentioned previously, multiple methods have been proposed for generating these replicate networks, and link inference algorithms have the potential to be used for this task. This research therefore answers the following questions.

1. Can link inference algorithms be adapted for effectively constructing network confidence intervals?
2. How do metric and network characteristics impact which methods yield the best confidence intervals?

In answering these questions, this chapter relies on simulated experiments conducted on stylized networks with edges missing completely at random (MCAR) [126, 166, 169], the models for which were introduced in Chapter 3.

4.2 Methods

4.2.1 Generating Simulated Networks

To evaluate the construction of confidence intervals for network metrics, hundreds of weighted, directed networks were generated in three styles: uniformly random, scale-free, and small-world. The networks were created using the same models specified in Chapter 3, with the notable exception that only single-layer networks were used for this work. For each style, networks of fifty or one hundred nodes were generated with densities of five percent or ten percent. One hundred networks were constructed for each possible configuration to be used as the true networks in subsequent experiments. Integer edge weights were then subtracted from each network, with the proportions removed set to either five percent, ten percent, twenty percent, or thirty percent. For each true network, this produced four observations with edges missing completely at random

(MCAR) [126, 166, 169] that could then be used to test the creation of confidence intervals for the metrics calculated on the true network.

4.2.2 Bootstrapping Networks

Five methods for bootstrapping networks were tested. All but one of these methods required the specification of the amount of edge information missing from the observed network, so three trials were conducted for each such bootstrapping model: one in which the model was provided with the precise total edge weight that had been removed from the true network, one in which the provided amount of missing edges was 75% of the accurate amount, and one in which the provided amount was 125% of the accurate amount. The bootstrapping method was then used to generate one hundred replicate networks from each observed network.

First, the Frantz and Carley [78] approach was tested by randomly augmenting the observed network with additional integer edge weights. This provided the most straightforward approach to bootstrapping networks and served as a sort of baseline against which to compare the other four methods.

Since these experiments were conducted on weighted, directed networks, the random dot product graph (RDPG) method advanced by Levin and Levina [123] required appropriate modification. This method relies on first computing the adjacency spectral embedding (ASE) of a network for a certain latent dimensionality d, which was adapted to directed networks by relying on singular value decomposition rather than eigendecomposition. The RDPG bootstrap then generates each replicate network by resampling the rows of the ASE with replacement, using the sampled rows to compute a probability matrix indicating the probability of existence for each edge, and conducting corresponding Bernoulli trials to generate a binary adjacency matrix. This procedure was modified to generate weighted networks by implementing the proposal by DeFord and Rockmore [60] to use the RDPG embedding matrix to parameterize Poisson distributions from which to draw the edge weights for the replicate network. In keeping with the work by Levin and Levina [123], the ASE dimensionality in this research was set to $d = 1$, though this introduces limitations discussed towards the end of this chapter.

Given that the RDPG approach described by Levin and Levina [123] was not explicitly designed to account for missing edges when fitting the RDPG model to the observed network, an additional version of this method was tested after implementing the masking idea proposed by Scheinerman and Tucker [170]. This called for the use of a binary mask matrix instructing the fitting algorithm to essentially pay less attention to the masked entries of the input adjacency matrix when constructing its embeddings [170]. Since network analysts often do not know which edges are missing in their data, a uniformly random mask matrix was created for generating each replicate network. Because this required fitting a new RDPG model for each replicate network, it added significantly to the computational expense of the process but was necessary in order to not assume which particular edges were missing from the network.

To test the possibility of using link inference for constructing network confidence intervals, two link inference algorithms were adapted for bootstrapping networks. Structural perturbation method (SPM) functions by repeatedly subsampling the input adjacency matrix, calculating the eigendecomposition of the subsampled matrix, using the removed edges to correct the eigenvalues (but not the eigenvectors), and recomposing the adjacency matrix using the corrected

eigenvalues [129]. This algorithm was later adapted to directed networks by subtracting the asymmetric edges from a network before calculating the eigendecomposition and then adding the edges back to the reconstructed adjacency matrix [211]. Rather than averaging together the reconstructed adjacencies over multiple repetitions of this process, each reconstructed matrix was taken as one of the replicate networks by rounding its entries to the nearest integer and zeroing any negative entries.

The same strategy was adopted with the link inference method proposed by Wang et al. [204], hereafter referred to as NMFP. This algorithm calls for adding noise to an input network (e.g., by randomly adding edges), compressing the noisy network into fewer dimensions via non-negative matrix factorization, and reconstructing a similar adjacency matrix from those weights and factors. Instead of averaging together the reconstructed matrices over multiple repetitions, each matrix was interpreted as its own replicate network.

Finally, it is worth noting that, except for the augmentation strategy of randomly adding edges, none of the tested bootstrapping methods are technically limited to being used solely on incomplete networks as opposed to networks with spurious connections. For the RDPG-based bootstraps, the process of drawing edge weights from parameterized Poisson distributions does not guarantee that edges present in the observed network will be present in the replicate networks. Similarly, the sampling and decomposition procedures used by SPM and NMFP can lead to the recomposed adjacency matrices containing low likelihood scores that would be remapped to zero. Indeed, Chapter 3 shows that SPM and NMFP can viably be used to identify spurious edges in a network. Hence, there is no technical reason for these bootstrapping methods to be limited to incomplete networks. However, the experiments presented in this chapter were only conducted on networks with missing edges, since they are of greater concern in the context of covert networks, and the ensuing results are therefore only reflective of incomplete networks.

4.2.3 Constructing Confidence Intervals

These five methods for generating replicate networks were used to construct 95% confidence intervals for three network-level and two node-level metrics. Each confidence interval was constructed by calculating the desired metric on each of the one hundred replicate networks, calculating the standard deviation of those "potential" metric values, and forming the 95% confidence interval centered at the value of the metric in the observed network. More specifically, for a metric m, observed network $\boldsymbol{O}$, and set of network replicates $\boldsymbol{N} = \{\boldsymbol{N}_1, \boldsymbol{N}_2, \ldots, \boldsymbol{N}_n\}$, the 95% confidence interval was calculated as $(m(\boldsymbol{O}) - 1.96\bar{\sigma}_{\boldsymbol{N}}, m(\boldsymbol{O}) + 1.96\bar{\sigma}_{\boldsymbol{N}})$, where $\bar{\sigma}_{\boldsymbol{N}}$ is the sample standard deviation of $\boldsymbol{N}$. This follows one of the procedures used by Levin and Levina [123], the limitations of which are discussed towards the end of this chapter.

4.2.4 Evaluating Interval Quality

The quality of the constructed confidence intervals was evaluated in terms of their coverages and widths. For each experimental configuration, coverage was calculated as the proportion of intervals that correctly contained the true network's metric value over the one hundred pairs of true and observed networks. This was averaged over all nodes for the node-level metrics. Since the experiments called for calculating 95% confidence intervals, the ideal coverage was 95%.

Though confidence intervals should be wide enough to reach their desired coverage, they should also be narrow enough to provide useful bounds for the metric being studied. The mean interval width was therefore calculated for each experimental configuration and normalized by the standard deviation of the true values for that metric across all networks of that same style. For node-level metrics, this normalization was conducted before averaging the interval widths by dividing each interval width by the standard deviation of the node metric values in the corresponding true network. This normalization process allowed the interval widths to be interpreted in terms of the variability of the metric being studied, thereby indicating whether the intervals were narrow enough to be useful for bounding the metric. Ideally, the intervals for network-level metrics would be narrow enough to distinguish one true network from another possible network, whereas the intervals for node-level metrics should ideally be narrow enough to not overlap with too many other nodes in the same network.

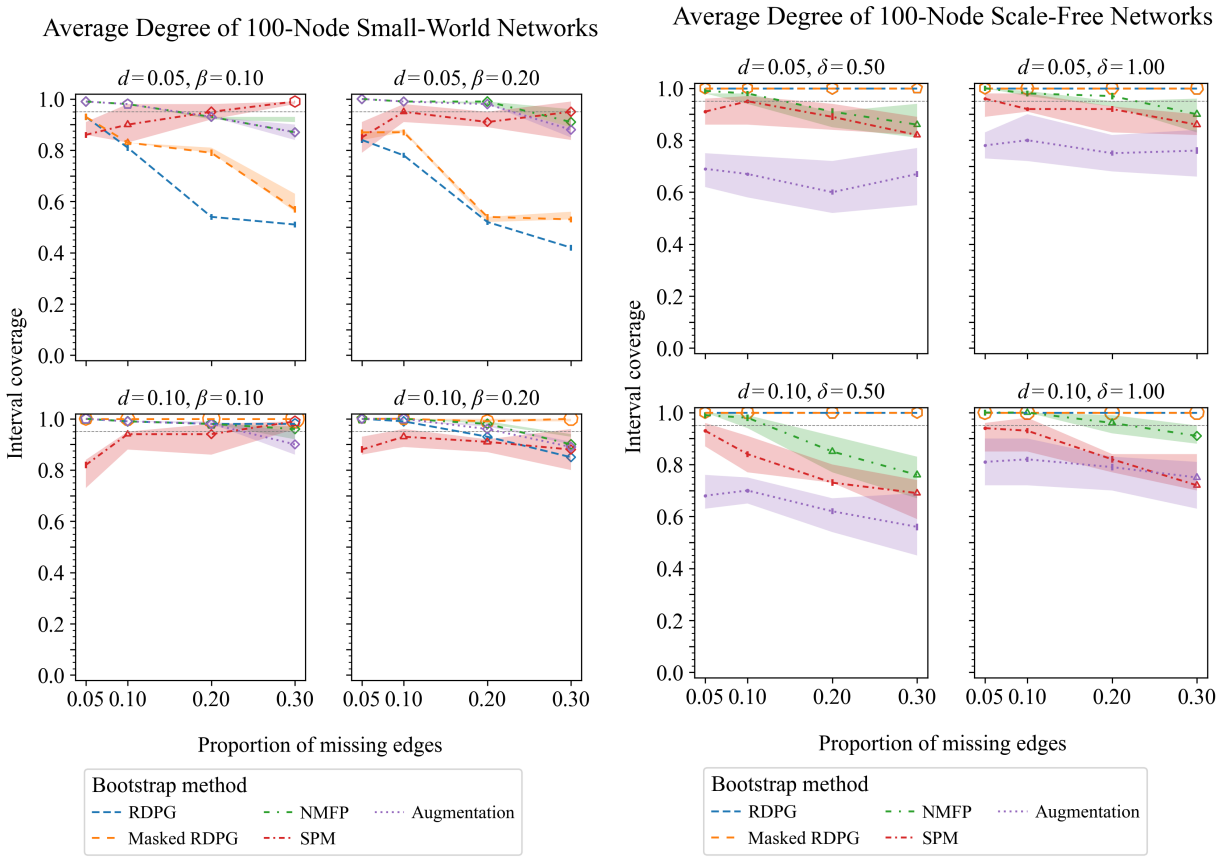
4.3 Results

4.3.1 Average Degree Centrality

Figure 4.1 shows the coverage from constructing confidence intervals for a network's average degree centrality. On small-world networks, coverage from the RDPG methods fell when more than 10% of the edges were missing, allowing the link inference algorithms to perform best. The opposite was true for scale-free and uniformly random networks (the latter of which are covered in Appendix G), with RDPG methods maintaining better coverage as more edges were removed. The masked version of the RDPG bootstrap tended to yield better coverage than the original method, and NMFP often outperformed SPM. Additionally, the sensitivity analysis (showed as shaded areas in Figure 4.1) revealed that the interval coverage of SPM was most vulnerable to changes in the assumption of the amount of missing data across all three network types, whereas NMFP and random augmentation exhibited high vulnerability only with scale-free and uniformly random networks.

Overall, the normalized interval widths show that the computed intervals were remarkably wide. For small-world networks with five percent of the edges missing, the methods achieving sufficient 95% coverage yielded intervals with widths less than four times the standard deviation of the average degree centralities for all hundred-node networks. The RDPG methods achieved coverages that were slightly too low but produced intervals of widths within two standard deviations of average degree centrality. This means that the large widths of the computed intervals would have made it difficult to distinguish between different possible true networks given an observed network. The results were slightly more promising for scale-free networks with up to ten percent of edges missing, for which SPM usually attained more than 90% coverage with intervals with widths within one standard deviation of the metric.

Comparing Figure 4.1 to the results for fifty-node small-world networks shown in Appendix G reveals that coverage was notably higher for the smaller networks and that interval widths were sometimes one standard deviation narrower for the smaller networks. Clear trends were less discernible when changing the size of scale-free networks. In terms of network configuration, comparing the top and bottom graphs in Figure 4.1a reveals that small-world networks of higher



(a) Small-world networks

(b) Scale-free networks

Figure 4.1: Confidence interval coverages for average degree centrality. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density, β being the probability of "long-distance" edges in small-world networks, and δ dictating the accumulation of edge weights in scale-free networks. Augmentation is the strategy of randomly adding edges.

densities tended to yield better confidence intervals, particularly with RDPG methods. This also held for uniformly random networks (as shown in Appendix G) but was not true for scale-free networks. Additionally, comparing the left and right graphs in Figure 4.1a shows that increasing the β parameter (i.e., the probability of "long-distance" edges) of the small-world model only resulted in slight decreases in performance under high observation error, though the statistical significance of this result is unclear. For scale-free networks, increasing the δ parameter (which increases edge weights) tended to increase confidence interval coverage.

4.3.2 Average Betweenness Centrality

Figure 4.2 shows the confidence interval coverage for a network's average betweenness centrality. Like for average degree, the adapted link inference algorithms outperformed RDPG methods on small-world networks, but this no longer applied to uniformly random networks, which are covered in Appendix G. The method of randomly adding edges also performed quite well on small-world networks, and scale-free networks again saw RDPG methods produce better coverage than link inference. Additionally, the masked RDPG method tended to perform better than the original RDPG bootstrap, and NMFP tended to outperform SPM. As with average degree centrality, the sensitivity analysis for the assumed amount of missing edges revealed large variability in interval coverage for SPM across all three network topologies and for NMFP and random augmentation on scale-free and uniformly random networks.

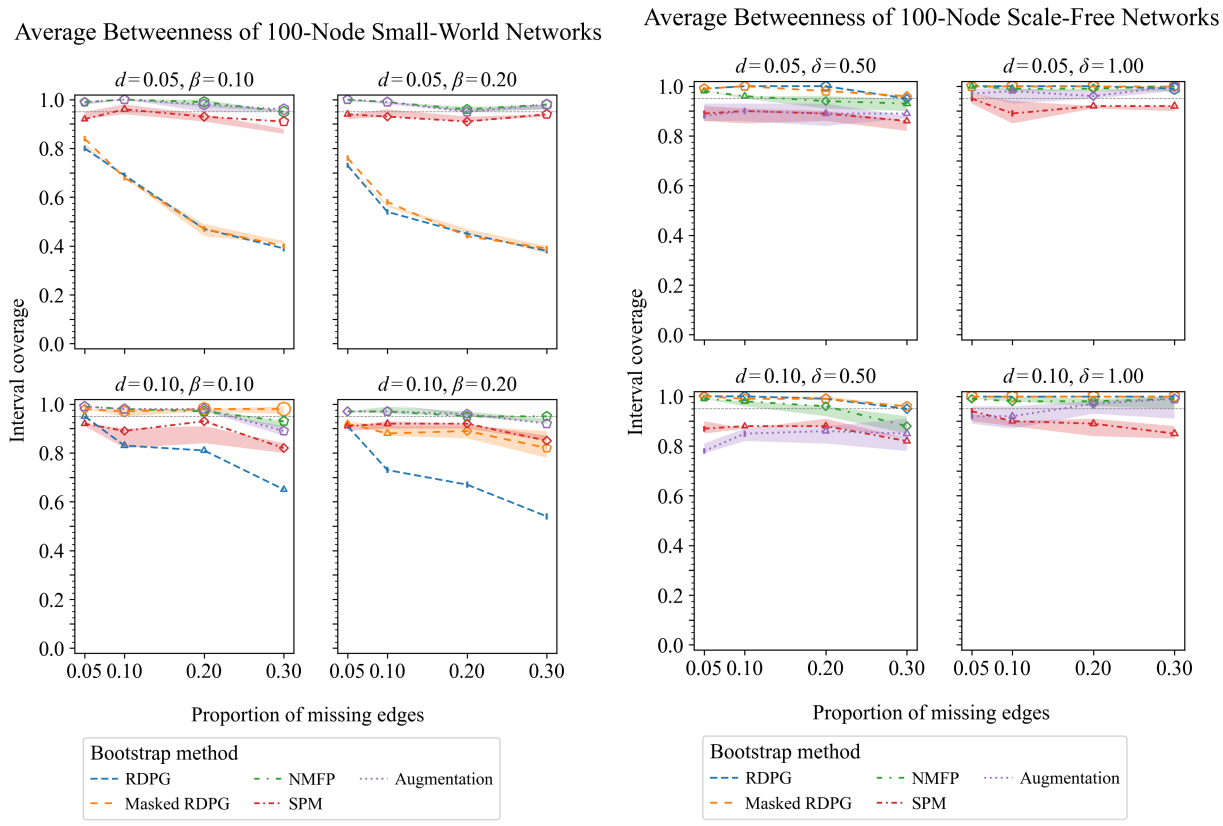
The normalized widths of the confidence intervals overall seemed to be approximately one standard deviation larger than those for average degree centrality. With five percent of edges missing in small-world networks, SPM almost attained sufficient coverage with intervals of widths less than three standard deviations of all hundred-node networks' average betweenness centralities. For scale-free networks, SPM and random augmentation sometimes attained greater than 90% coverage with intervals as wide as two standard deviations of the metric.

Comparing Figure 4.2 to the results for fifty-node networks in Appendix G shows that interval coverage improved significantly when increasing the size of small-world networks (particularly for the RDPG methods), whereas slight drops in coverage could be seen when adding more nodes to the scale-free networks. In terms of other parameters, greater density in small-world and random networks tended to yield better intervals, but not in scale-free networks. Increasing the β parameter in small-world networks did not discerningly impact interval coverage, but increasing the δ parameter in scale-free networks produced slight increases in performance.

4.3.3 Triangle Density

Figure 4.3 shows the confidence interval coverage for a network's triangle density. These results differ significantly from those for average degree and average betweenness, since the link inference methods often failed to provide even moderate coverage. Except on uniformly random networks, the RDPG methods were able to reach satisfactory coverage at the expense of providing intervals with widths several times larger than the standard deviation of the true triangle densities. For small-world networks, the unmasked RDPG bootstrap usually yielded narrower (and therefore more useful) confidence intervals while maintaining adequate coverage. On uniformly random networks, the RDPG methods only yielded good coverage when at most 10% of the edges were missing. Finally, the sensitivity analysis again showed large variation in the coverage of SPM, with NMFP and augmentation showing more variation for scale-free networks than small-world networks.

Comparing Figure 4.3 to the fifty-node results in Appendix G again reveals that increasing the size of small-world networks tended to increase interval coverage, whereas the opposite was true for scale-free networks. With regard to network configuration, few trends appeared. For small-world networks, increasing the probability of "long-distance" edges decreased the interval coverage for link inference methods, but it also narrowed the widths of the satisfactory intervals



(a) Small-world networks

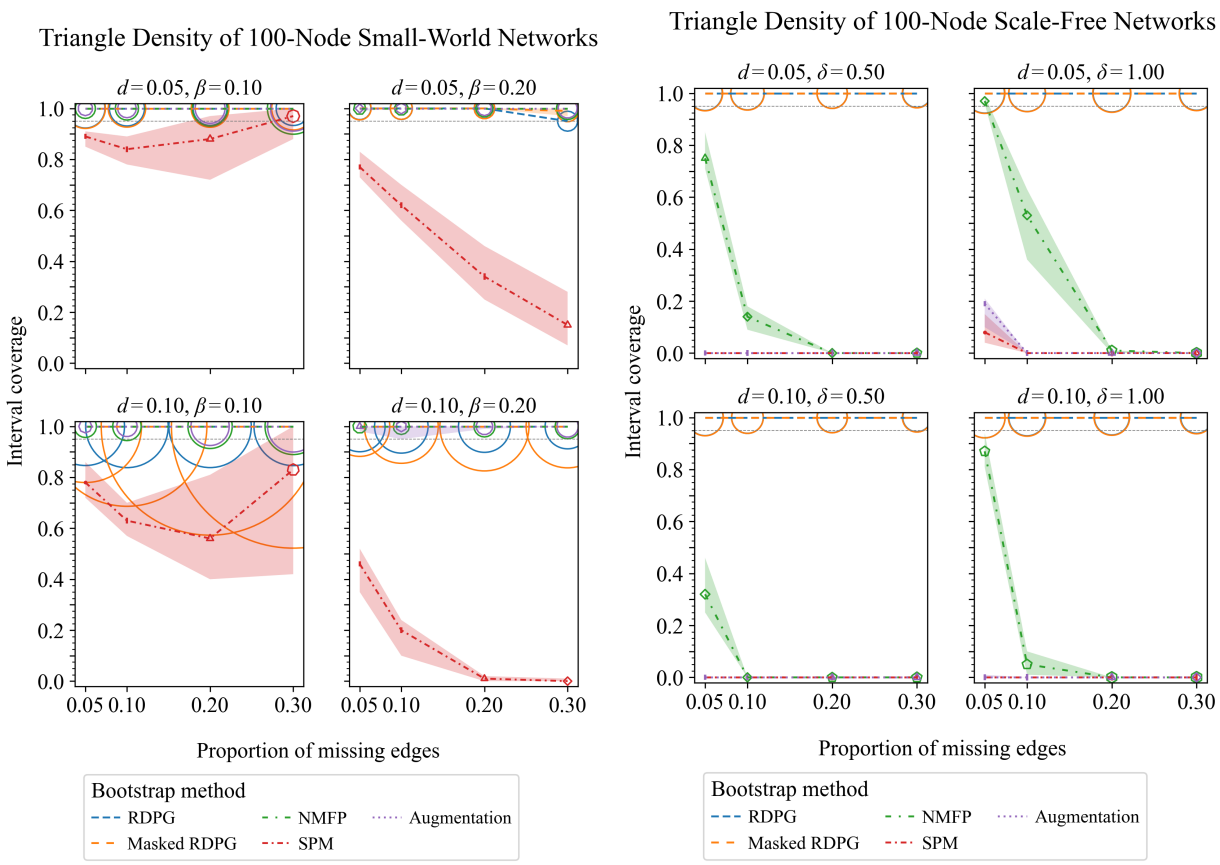
(b) Scale-free networks

Figure 4.2: Confidence interval coverages for average betweenness centrality. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density, β being the probability of "long-distance" edges in small-world networks, and δ dictating the accumulation of edge weights in scale-free networks. Augmentation is the strategy of randomly adding edges.

produced by the RDPG methods. For scale-free networks, increasing the overall edge weights tended to improve the performance of NMFP, but it still only reached adequate coverage when five percent of the edge information was missing.

4.3.4 Degree Centralities

Figure 4.4 shows the confidence interval coverage for node-level degree centralities. Four of the five methods attained sufficient coverage across all proportions of missing edges for small-world and scale-free networks, and SPM almost always achieved at least 90% coverage. The results for uniformly random networks provided in Appendix G show slightly lower coverage for the



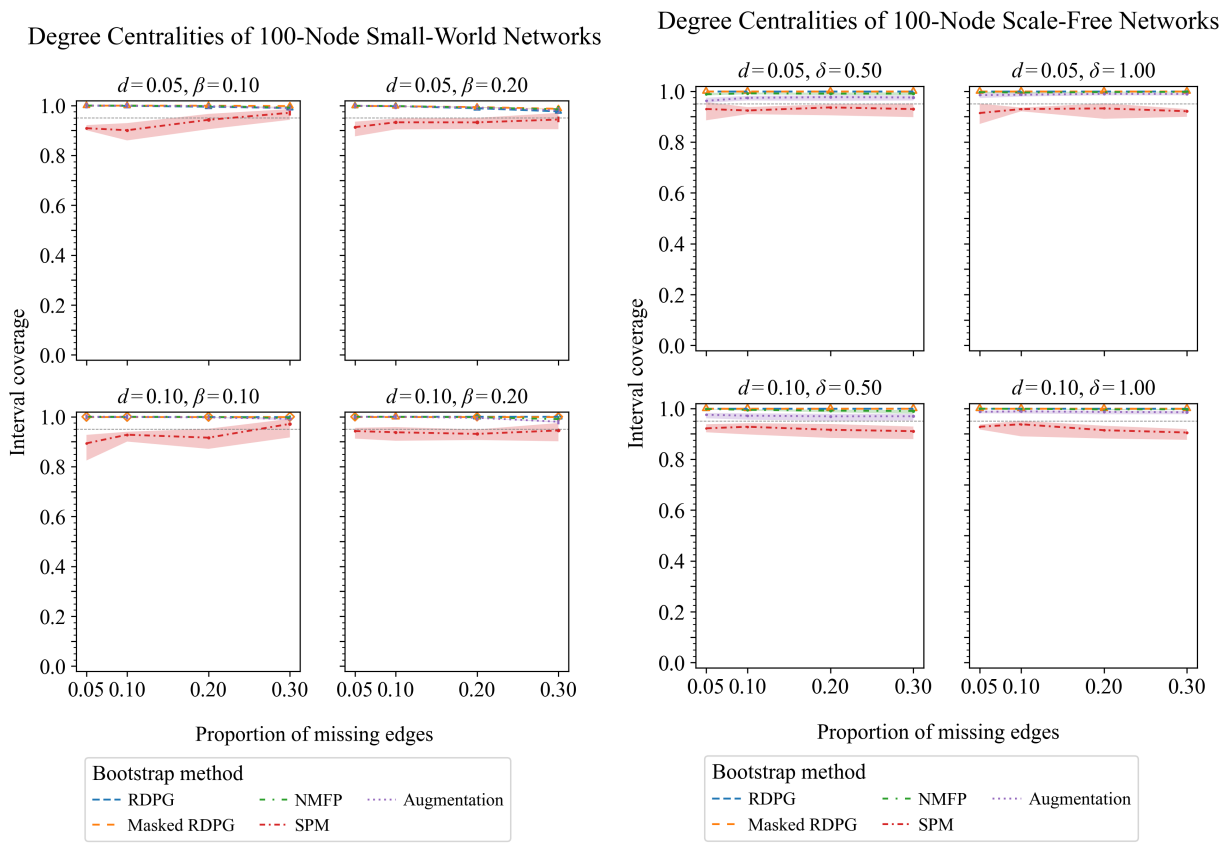
(a) Small-world networks

(b) Scale-free networks

Figure 4.3: The confidence interval coverages for triangle density. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density, β being the probability of "long-distance" edges in small-world networks, and δ dictating the accumulation of edge weights in scale-free networks. Augmentation is the strategy of randomly adding edges.

non-RDPG methods, but they all approached or exceeded 90%. In terms of the assumption on the amount of missing edges, SPM was again the most sensitive, with random augmentation also showing some sensitivity for scale-free networks. However, since many methods achieved close to 100% coverage with large intervals, it may be difficult to notice differences in algorithm sensitivity to this parameter.

The normalized widths of these node-level intervals were much narrower than for the network-level metrics. The strategy of randomly adding edges to the observed network was always able to achieve sufficient coverage except in the uniformly random networks and did so with intervals smaller than one standard deviation of the nodes' degree centralities. This means that the



(a) Small-world networks

(b) Scale-free networks

Figure 4.4: The confidence interval coverages for node-level degree centralities. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density, β being the probability of "long-distance" edges in small-world networks, and δ dictating the accumulation of edge weights in scale-free networks. Augmentation is the strategy of randomly adding edges.

intervals maybe have been useful at accurately distinguishing high-centrality nodes from low-centrality nodes in the true, unseen network.

Comparing Figure 4.4 to the fifty-node results in Appendix G shows some slight decreases in coverage for the smaller networks, but the differences were small. Changes in network density did not seem to noticeably impact interval coverage, whereas increasing the accumulation of edge weights in scale-free networks appeared to slightly boost interval coverage for random augmentation.

4.3.5 Betweenness Centralities

Figure 4.4 shows the confidence interval coverage for node-level betweenness centralities. For small-world networks with one hundred nodes, NMFP and random augmentation achieved sufficient coverage when up to twenty percent of edges were missing, but this came at the cost of interval widths within four standard deviations of the nodes' betweenness centralities. In contrast, scale-free networks saw the RDPG methods consistently attaining sufficient coverage, whereas the other methods eventually dropped below the 95% mark as more edges were removed during the observation process. However, NMFP was able to achieve sufficient coverage with up to ten percent missing edges via intervals with widths less than one standard deviation of the nodes' betweenness centralities. Finally, the sensitivity analysis generally matched the results for the node-level degree centralities.

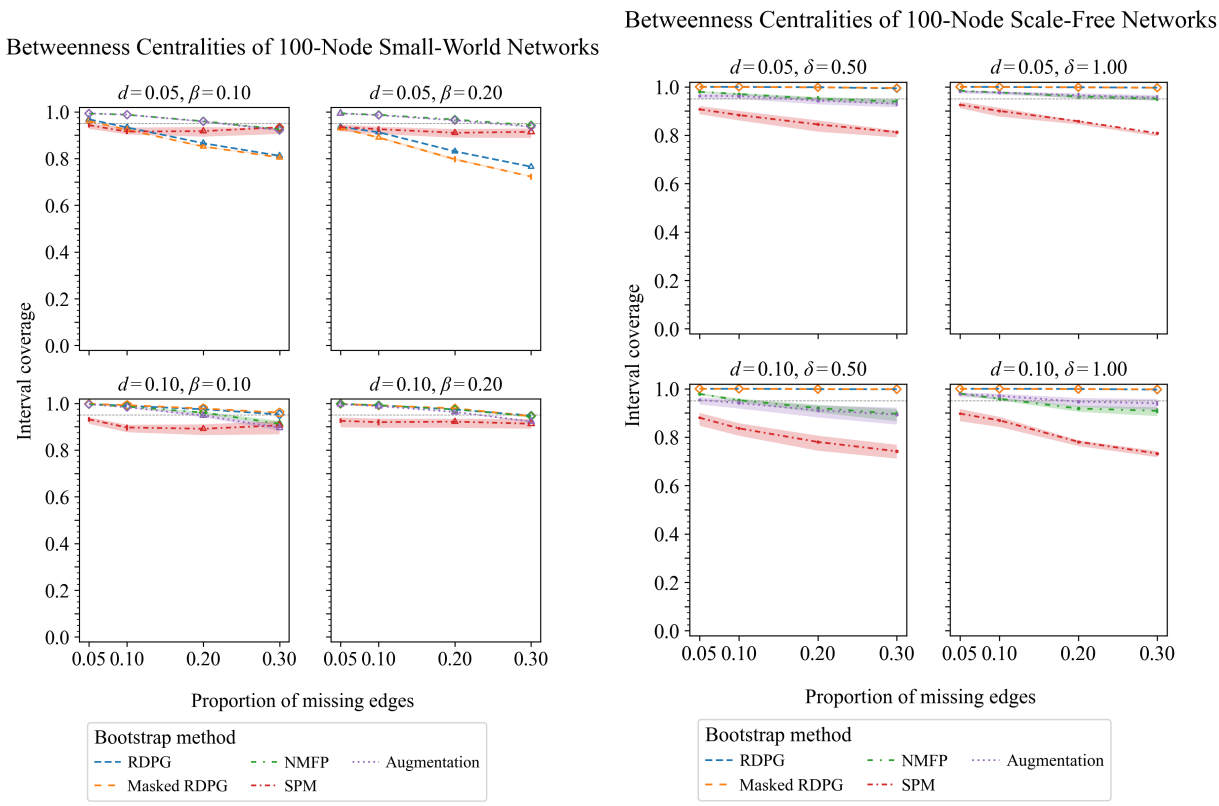
Comparing Figure 4.4 to the results for fifty-node networks in Appendix G shows the coverage of RDPG-based intervals being significantly lower for smaller small-world networks, whereas the coverage of non-RDPG methods suffered on smaller scale-free networks. Figure 4.4 also shows that increasing the density of small-world networks tended to increase interval coverage, whereas increasing the density of scale-free networks tended to slightly decrease coverage. Increasing the probability of "long-distance" edges in small-world networks did not seem to have a clear impact on interval coverage, but increasing the accumulation of edge weights in scale-free networks appeared to minimally increase coverage.

4.3.6 Runtime

Figure 4.6 shows the execution times for using the various bootstrapping approaches to generate one hundred replicates for small-world and scale-free networks. These results show that the masked RDPG bootstrap was the most computationally expensive approach for small-world networks. This was to be expected, given that a new RDPG model must be fit to generate each replicate network and that the convergence time of the masked embedding process is highly variable depending on the particular random mask that is generated. In contrast, NMFP took the longest to run on scale-free and uniformly random networks, as shown in Appendix G.

4.4 Discussion

Sadly, the main finding of these experiments was that the tested bootstrapping methods usually yielded confidence intervals that either had insufficient coverage for incomplete networks or were exceedingly wide. This comes in contrast to the sufficient coverage achieved by Levin and Levina [123], who were able to guarantee proper coverage on binary networks observed without error. This work instead considered weighted networks with missing edges, making the problem significantly more difficult. Additionally, Levin and Levina [123] generated the networks in their study using RDPG models and also used RDPG models to subsequently construct their confidence intervals. Because the present work instead generated the original networks using traditional stylized network models and tested multiple bootstrapping methods based on various underlying models, the guarantees achieved by Levin and Levina [123] no longer applied.

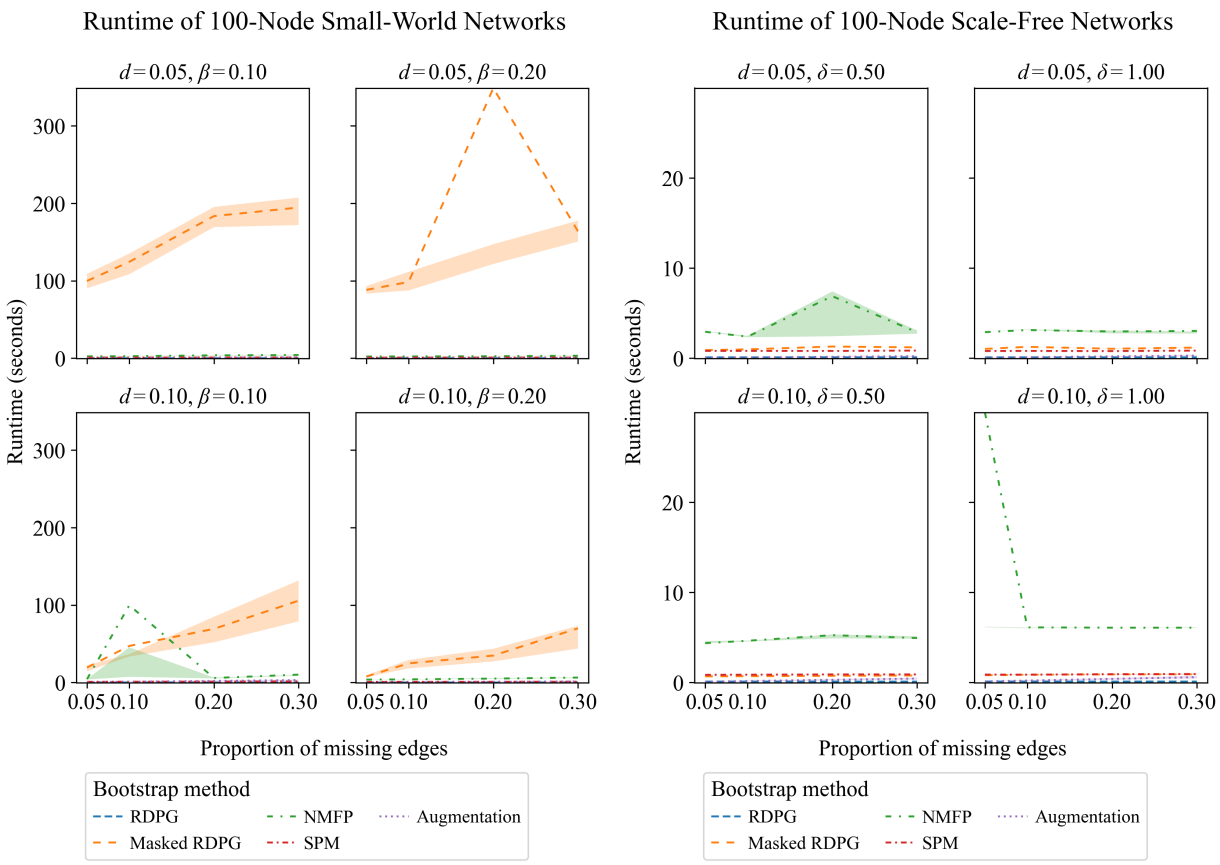


(a) Small-world networks

(b) Scale-free networks

Figure 4.5: The confidence interval coverages for node-level betweenness centralities. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density, β being the probability of "long-distance" edges in small-world networks, and δ dictating the accumulation of edge weights in scale-free networks. Augmentation is the strategy of randomly adding edges.

Several useful commonalities also emerged across these experiments. In terms of bootstrapping methodology, the adapted link inference algorithms tended to produce the best confidence intervals when not dealing with scale-free networks or triangle density, for which the RDPG methods performed best. Moreover, the masking process often improved the performance of the RDPG bootstrap. However, it is important to note that only the standard RDPG bootstrap does not require the user to specify the number of missing edges. The other methods require analysts to estimate how much error is present in their network data, which can be a significant limitation. The sensitivity analysis revealed that the non-RDPG bootstrap methods exhibited greater variability in coverage when changing the assumed amount of missing edges, with SPM being particularly sensitive.



(a) Small-world networks

(b) Scale-free networks

Figure 4.6: The runtimes for bootstrapping small-world and scale-free networks. Shaded areas indicate the spread in each algorithm's runtime when the assumption of the amount of missing data is 25% too low or 25% too high. Each subplot presents a different network configuration, with d indicating network density, β being the probability of "long-distance" edges in small-world networks, and δ dictating the accumulation of edge weights in scale-free networks. Augmentation is the strategy of randomly adding edges.

In terms of network characteristics, small-world networks tended to be the most challenging for generating good confidence intervals. Additionally, increasing the density of small-world and uniformly random networks yielded intervals with better coverage, and increasing the overall edge weights of scale-free networks generally led to better intervals. These results suggest that networks that already contain less randomness and more edge information (either in terms of density or weight) are better for generating accurate confidence intervals for network metrics. Perhaps unsurprisingly, this matches closely with the conclusions drawn from the link inference analysis in Chapter 3.

In terms of network metrics, average degree centrality and average betweenness centrality showed similar results, whereas triangle density was much more difficult to bound and required RDPG bootstrapping methods. This is likely because triangles are easily broken by missing edges, whereas degree and betweenness are less influenced by any single edge or path, respec-

tively. Notably, the experiments on node-level metrics yielded coverages that were more stable over different proportions of missing edges than those for the network-level measures. The interval widths also appeared to be comparatively narrower for the node-level metrics, though the normalization process was different and the intervals for the node-level centralities were still too wide to avoid significant overlap between nodes. The more stable performance on node-level metrics is an encouraging finding for analysts studying a single covert network, since they are likely to be more concerned with distinguishing between nodes within a network than with estimating a network-level property.

4.5 Conclusion

4.5.1 Limitations

The experiments presented in this chapter suffer from several limitations, chiefly the exclusive reliance on simulated networks with uniformly random error. Conducting experiments using real network data and more realistic observation processes replicating other types of missingness [126, 166, 169] would more readily demonstrate the effectiveness of these confidence interval methods when used in practice, but a theoretical comparison of their performance is a significant first step in understanding their viability for use on incomplete networks. Additionally, certain particularities of the methodology likely impacted the results of these experiments. Though Levin and Levina [123] achieved good coverage results using a latent dimensionality of one for the RDPG bootstrap, it is possible that using an algorithm to automatically select the latent dimensionality could have resulted in confidence intervals with different coverages or widths. Except on certain small-world networks, the experiments presented in this chapter also showed relatively good coverage for RDPG-based methods, but further research on the impact of the chosen latent dimensionality would be helpful for practitioners wishing to optimize the performance of these methods. Another significant limitation to this work is its reliance on a normal distribution when constructing the final confidence interval based on the sample standard deviation of the replicate metrics. Not all network metrics have a normal limiting distribution [134], which may result in inaccurate confidence intervals. Forming percentile-based confidence intervals did not yield better results than those presented in this chapter, but further research on this matter might establish better ways of forming confidence intervals for network metrics calculated on incomplete networks.

4.5.2 Contributions

This chapter compares the use of various network bootstrapping methods on incomplete networks for constructing confidence intervals for network metrics. Rather than applying bootstrapping to a complete network to attain confidence intervals relevant to a population of networks [86, 123], this work demonstrates the use of bootstrapping on networks with missing edges to attain confidence intervals for the unseen complete network. Unfortunately, this process was sometimes not able to attain the desired coverage for network-level metrics, whereas node-level metrics fared better.

This comparison of bootstrapping methods also instructs network analysts on which approaches to use in certain situations. The link inference methods yielded better interval coverage for small-world and uniformly random networks, except on triangle density. Conversely, the RDPG-based methods worked best on scale-free networks and triangle density. The sensitivity analysis also highlighted that the link inference methods provided more variable coverage depending on the assumed amount of missing edges, so an analyst may want to opt for an RDPG-based method if the amount of missing edges is highly uncertain. Moreover, the results showed that analysts are most likely to get good confidence intervals for scale-free networks, which is particularly helpful for social media networks [7, 188].

Significantly, this work demonstrates that link inference algorithms can be used for bootstrapping networks to construct confidence intervals for network metrics. This should open up new avenues for exploration in future research on this topic. Indeed, this chapter points to the need for additional research on bounding metrics calculated on incomplete networks, since the tested bootstrapping approaches frequently had insufficient coverage or produced particularly large intervals. Further research may identify better methods or ways of using the confidence intervals despite their weaknesses.

Chapter 5

Improving Analytical Robustness[1]

5.1 Introduction

5.1.1 Motivation

The preceding three chapters have developed and evaluated three approaches to dealing with absent or imperfect information in social networks. However, the ultimate goal of bettering the forensic analysis of such networks requires that these methods improve the forms of enquiry that network analysts might want to explore with their data, such as identifying key actors. Given that past studies have shown how vulnerable node centrality measures are to missing information [28] or temporal changes [108], using techniques like link inference should make such analysis more accurate. Unfortunately, little attention has been paid to this matter, with link inference studies almost exclusively evaluating their algorithms' direct performance at identifying unseen edges rather than also considering downstream tasks.

Moreover, the techniques outlined in this thesis could be used to actively prevent missing data by improving the collection of network data in the first place. Since data gathering must often be carried out with finite funds or capabilities, it is imperative for those resources to be allocated in a manner that optimizes the collection of relevant information. Though a chief example for covert networks would be law enforcement's need to appropriately target their limited surveillance resources, this issue is also particularly important in social media analysis, since data collection can face severe restrictions from platforms that limit both the amount of data that can be gathered and the sampling process used to select those data [140, 199]. Enhancing the collection of incomplete network data is therefore an integral part of coping with that incompleteness.

5.1.2 Objectives

This chapter explores how the approaches developed in this thesis might improve covert network analysis by applying them all to the same data set, namely the French terror tweets described in Chapter 3. More specifically, this work evaluates the extent to which this thesis's techniques can

[1] Parts of this chapter were presented at the 2023 Informed Democracy and Social-cybersecurity Conference [14].

improve the calculation of centrality metrics on incomplete networks and the collection of relevant social media data. To accomplish these goals, this chapter answers the following questions.

1. How does link inference impact the accuracy of centrality measures calculated on an incomplete network?
2. How does link prediction impact the stability of centrality measures over time?
3. Can creation bursts identify promising accounts for collecting more nefarious social media activity?
4. How does collecting more data about specific actors impact the uncertainty associated with their centralities?

The remainder of this chapter proceeds by exploring each question in order before jointly discussing the resulting implications.

5.2 Improving the Accuracy of Node Centralities

5.2.1 Background

Given that missing data can lead network analysts to reach inaccurate conclusions [28, 76, 140, 199], it is unfortunate that little research has studied how the robustness of final analysis conducted on incomplete networks might be improved. Though significant work has gone into revealing hidden edges in a network [118, 128] or imputing missing data when fitting a network model [115, 116], hardly any research has investigated whether these techniques for filling in missing network data can improve the reliability of downstream analysis conducted using other methods. The creators of a network reconstruction algorithm stated that their reconstructed network qualitatively appeared to better capture nodes' degree and betweenness centralities [91], and a simulation study in bioinformatics concluded that link prediction can increase the quality of protein interaction networks [100]. In terms of covert networks, a study using mafia data found that predicting links in simulated smaller networks could decrease their divergence from the original network [38], but further analysis of the resulting networks was not considered. Importantly, it seems that no research using non-simulated networks has been conducted on this issue. Rather than simply assuming that techniques for reconstructing missing network data can improve downstream analysis, a closer examination of this hypothesis should be performed.

5.2.2 Methods

Network Augmentation

To test whether centrality measures calculated on an incomplete network become more accurate after adding inferred links, experiments were run using the dynamic and multiplex communication network formed from the streamed sample of French terror tweets described in Chapter 3. The edge likelihood scores generated by TPM on this network were summed together across all layers and time periods to attain a single likelihood score for each pair of users. It was then necessary to choose how many of the top-ranking links to add to the sample network. Because this choice could impact the results, this experiment was repeated for varying numbers of edges

added to the sample network, ranging from the equivalent of one quarter of the edges already in the sample network to twice the number of edges in the network.

Measurement of Centrality Accuracy

After augmenting the sample network with edges inferred by TPM, various node centrality measures were calculated on both the original sample network and the multiple augmented networks. Since these were multidimensional networks, their layers and time slices were summed together before calculating the weighted versions of degree, eigenvector, betweenness, and closeness centrality. All metrics were normalized to range from zero to one by dividing by their maximum value, and weighted closeness centrality was calculated by inverting the edge weights (since closeness interprets them as distances).

The accuracy of these node centralities was then evaluated by comparing them to the same centrality measures calculated on the similarly summed timeline network described in Chapter 3. This comparison was first conducted by calculating the Kendall rank correlation between the sample and timeline networks and between the augmented and timeline networks. Following Borgatti et al. [28], the precisions (i.e., true positive rates) for correctly identifying the top-ranking nodes were also calculated for the top ten percent of nodes (i.e., top one hundred nodes) and the top one percent of nodes (i.e., top ten nodes).

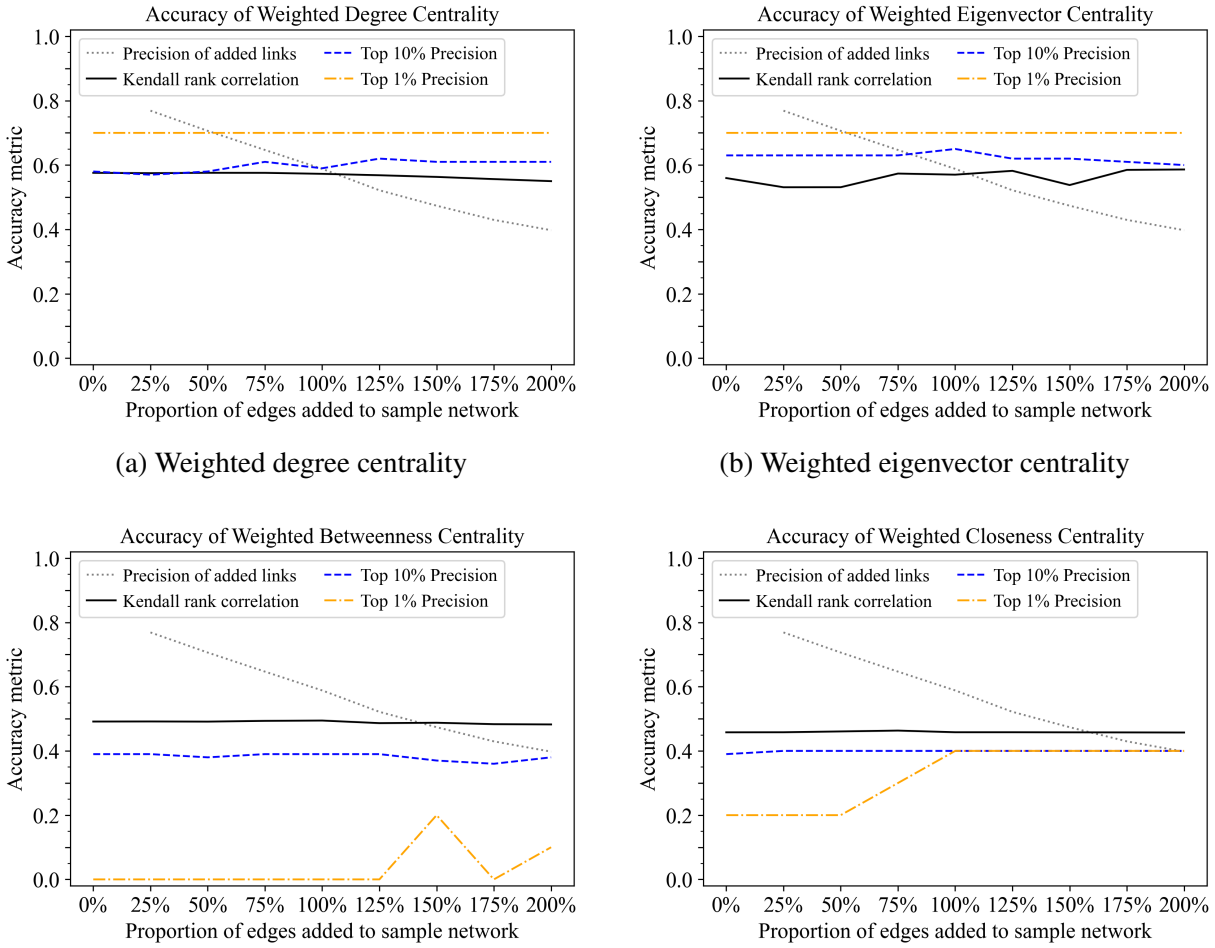
5.2.3 Results

Figure 5.1 shows the accuracy results for the centrality measures calculated on the original sample network (located at the far left with no edges added) and on the various augmented networks. The results for degree centrality show that it was somewhat accurate even without using link inference, with approximately sixty of the top one hundred nodes and seven of the top ten nodes identified correctly. Unfortunately, the results also show that degree centrality exhibited only slight improvement in identifying the top one hundred nodes when adding predicted links. Eigenvector centrality performed roughly equivalently to degree centrality without using link inference but showed only slight changes after adding inferred links.

The results for betweenness centrality show that it was significantly less accurate than degree or eigenvector centrality when not using link inference, as it was not able to correctly identify any of the top ten nodes. Adding inferred links was sometimes able to improve the results, helping it correctly identify two of the top ten actors when adding 150% more links to the network. The raw performance of closeness centrality was slightly better, with it being able to correctly identify two of the top ten nodes. Notably, adding inferred links doubled the top one percent precision after adding 100% more links.

5.2.4 Discussion

Overall, it is clear that it was extremely difficult for link inference to improve the accuracy of centrality measures based on connection counts, such as degree and eigenvector centrality. However, these were already surprisingly accurate when considering that the raw sample network contained only 9.2% of all the edges in the timeline network. Even making the sample network



(a) Weighted degree centrality

(b) Weighted eigenvector centrality

(c) Weighted betweenness centrality

(d) Weighted closeness centrality

Figure 5.1: The accuracy of various node centralities measured against the French terror timeline network when adding varying proportions of inferred links. Accuracy on the original sample network is shown at the far left with no links added, and the precision of the inferred links is shown in the background as a point of reference.

150% more dense using inferred links would still give it less than a quarter of the number of edges in the timeline data, partly explaining why it is so difficult to sway the node centrality rankings. In contrast, link inference was sometimes able to help the centrality measures based on shortest paths, like betweenness and closeness centrality. This makes intuitive sense, since even a single new edge can make a large difference in the shortest paths running through a given node. Though it is not clear what level of accuracy would be required by practitioners, these accuracy figures may help intelligence analysts determine their level of analytic confidence in their conclusions, a topic that has received relatively little quantitative research [157].

	T_1	$T_1 + T_2$	$T_1 + T_2 + T_3$	$T_1 + T_2 + T_3 + T_4$
Unweighted density	0.00279	0.00495	0.00647	0.00710
Total weight	38913	81972	123152	152709

Table 5.1: Densities and total weights for the cumulative time periods formed from the French terror tweets described in Chapter 3. T_w represents the adjacency matrix for communication between users during week w.

5.3 Stabilizing Node Centralities Over Time

5.3.1 Background

Aside from being vulnerable to edges missing from past data, centrality measures are also known to vary over time [108], making it difficult to accurately identify central actors when a significant amount of the activity under study has yet to occur. This can be particularly problematic for analyzing social media, since new messages are constantly being published. Several scholars have proposed centrality measures specifically for dynamic networks [71, 122, 186], but these have been designed to address the potential for an actor's underlying importance to change over time rather than to improve the quantification of an actor's importance when little information is available. Some work has tried reformulating centrality measures to make them more stable in the face of noise added to a network [172], but link prediction has the potential to improve the stability of existing measures by filling in some of the as-yet-unknown information.

5.3.2 Methods

Dynamic Network Augmentation

To test whether adding predicted links to a network could produce more stable node centralities as more data are collected, another experiment was conducted using the four-week dynamic multiplex network formed from the French terror tweets defined in Chapter 3. To simulate the process of iteratively accumulating more social media data, each of the four time slices was summed with all its preceding time slices. In other words, this resulted in four networks summarized in Table 5.1: the first week on its own, the sum of the first two weeks, the sum of the first three weeks, and the sum of all four weeks. Each of these networks was run through TPM, and the likelihood scores were summed across the two layers, since this experiment did not need to differentiate between different types of interactions between users. These separate sets of likelihood scores were then used to add predicted links to each of the four networks. Since the statistics in Table 5.1 indicated that the increase in total weights was around 100% for the first transition, 50% for the second transition, and 25% for the last transition, the process of predicting links was conducted twice, first by consistently adding 25% more links to each network and then by consistently adding 100% more links to each network.

Measurement of Centrality Stability

After augmenting each time slice of the sample network with edges predicted by TPM, various node centrality measures were calculated on the original sample networks and the multiple augmented networks. To do this, the two layers of the networks were summed together, and the weighted versions of degree, eigenvector, betweenness, and closeness centrality were then calculated as in the previous experiment. The stability of these centralities was then measured by comparing consecutive time periods in terms of their Kendall rank correlation, Jaccard index [103] among the top ten percent of nodes, and Jaccard index among the top one percent of nodes. Additionally, the Segarra-Ribeiro K measure of centrality stability [173] was also calculated, which defines the stability of a centrality metric as a constant that, when multiplied by the total difference between two adjacencies, bounds the maximum change in centrality between the networks. To understand whether adding predicted links improved centrality stability, this comparison was performed three times: first between consecutive raw networks, then between each network augmented with 25% more links and its subsequent raw network, and finally between each network augmented with 100% more links and its subsequent raw network.

5.3.3 Results

Table 5.2 shows the stability measures for the weighted centrality measures. Without applying link prediction, betweenness centrality was the least stable metric across all stability measures, which matches previous findings [173]. As expected, stability improved as more data were added, with the last transition to the fourth time period achieving high stability scores, including Kendall rank correlations ranging from 0.87 to 0.96 over all centralities and complete Jaccard similarity for degree, eigenvector, and closeness centrality.

Unfortunately, adding predicted links largely did not improve centrality stability. Betweenness centrality in particular became exclusively less stable across all measures of stability, with Jaccard similarity for the top one hundred nodes decreasing by 0.22 over the last temporal transition. Closeness centrality showed almost no change from applying link prediction, whereas degree and eigenvector centrality showed very slight improvements in top ten percent Jaccard similarity over the first temporal transition when adding 100% more links. However, these increases in stability over the first transition were coupled with smaller decreases in stability over the second transition. Since the first temporal transition corresponded to collecting approximately 100% more data and the second transition would have required adding fewer predicted edges, it seems that adding the correct amount of predicted edges may be important for whether link prediction improves centrality stability.

5.3.4 Discussion

Overall, these results show that applying TPM did not improve the stability of centrality measures over time. A different link prediction algorithm may have fared better, so it is possible that additional research might achieve better results. More research would also be helpful for understanding the impact of the number of predicted edges added to the input networks and how that might be adjusted over time. Nonetheless, it is encouraging to see that the node centralities

Weighted Centrality	Original Sample I	Original Sample II	Original Sample III	Change After Adding 25% More Links I	Change After Adding 25% More Links II	Change After Adding 25% More Links III	Change After Adding 100% More Links I	Change After Adding 100% More Links II	Change After Adding 100% More Links III
Degree	0.790	0.884	0.955	+5e−5	+2e−5	−2e−4	−1e−3	−0.002	−0.006
Eigenvector	0.725	0.834	0.932	−0.002	−0.021	−1e−3	+0.013	+0.002	−0.005
Betweenness	0.632	0.752	0.873	−0.001	−5e−5	−0.001	−0.004	−0.029	−0.080
Closeness	0.740	0.832	0.924	+0.001	+0.002	+4e−4	−0.002	−0.001	−0.006

(a) Kendall rank correlation

Weighted Centrality	Original Sample I	Original Sample II	Original Sample III	Change After Adding 25% More Links I	Change After Adding 25% More Links II	Change After Adding 25% More Links III	Change After Adding 100% More Links I	Change After Adding 100% More Links II	Change After Adding 100% More Links III
Degree	6e−6	6e−6	2e−6	+5e−8	+1e−7	−1e−8	−2e−7	−9e−8	−2e−7
Eigenvector	9e−6	7e−6	7e−6	+1e−7	+2e−8	+7e−8	−1e−7	−3e−7	−8e−7
Betweenness	8e−6	1e−5	1e−5	+1e−7	+2e−7	+3e−7	+5e−6	+1e−5	+9e−6
Closeness	9e−5	1e−4	7e−5	+1e−6	+2e−6	+1e−6	+1e−6	−1e−6	−5e−6

(b) Segarra-Ribeiro K (lower values indicate greater stability) [173]

Weighted Centrality	Original Sample I	Original Sample II	Original Sample III	Change After Adding 25% More Links I	Change After Adding 25% More Links II	Change After Adding 25% More Links III	Change After Adding 100% More Links I	Change After Adding 100% More Links II	Change After Adding 100% More Links III
Degree	0.64	0.82	0.83		−0.02		+0.03	−0.02	
Eigenvector	0.55	0.74	0.87				+0.02	−0.01	
Betweenness	0.44	0.59	0.69		−0.01			−0.12	−0.22
Closeness	0.80	0.82	0.94						

(c) Top 10% Jaccard index [103]

Weighted Centrality	Original Sample I	Original Sample II	Original Sample III	Change After Adding 25% More Links I	Change After Adding 25% More Links II	Change After Adding 25% More Links III	Change After Adding 100% More Links I	Change After Adding 100% More Links II	Change After Adding 100% More Links III
Degree	0.54	0.82	1.00						
Eigenvector	0.82	1.00	1.00						
Betweenness	0.25	0.33	0.33					−0.16	
Closeness	0.54	0.54	1.00						

(d) Top 1% Jaccard index [103]

Table 5.2: Measures of centrality stability calculated on consecutive pairs of networks before and after inferring links. Roman numerals indicate different temporal transitions: I is the transition between week one and week two, II is the transition from the sum of weeks one and two to week three, and III is the transition from the sum of weeks one, two, and three to week four.

other than betweenness tended to stabilize on their own as more data were collected, particularly for the top one percent of nodes.

5.4 Identifying Subjects for Data Collection

5.4.1 Background

Prior research on social media data collection has mostly focused on developing good search terms and filters. A few scholars have proposed models for more rigorous collection processes, with Kim et al. [111] calling out what they saw as poor collection practices in digital health research and proposing the use of precision and recall metrics for quantifying the quality of social media data sets. In terrorism research, Parekh et al. [154] instead advocated for a multi-stage data collection process, starting on a set of seed accounts and iteratively filtering and expanding that collection before verifying the quality of the final data set. Such a multi-stage process could be used to bias the data towards specific qualities, with secondary stages dedicated to collecting more data reflecting behaviors of interest, such as automated activity or misinformation sharing. If the goal is indeed to increase the collection of concerning social media activity, the results of Chapter 2 suggest that identifying accounts created during bursts could be an effective way to select subjects for further data collection.

5.4.2 Methods

Burst Detection

To investigate whether account creation bursts could identify users exhibiting more concerning activity in the French terror data, the method described in Chapter 2 was employed to study the creation times of accounts featured in the original streamed sample of French terror tweets, which is described in Chapter 3. Figure 5.2 shows the dates that were detected as spikes in account creations using precisely the same method used in Chapter 2. At a probability of false alarm (PFA) of 0.25, 395 days were detected as bursts, whereas a PFA of 0.30 identified 829 bursts and a PFA of 0.35 identified 1827 bursts.

Quantification of Concerning Activity

Given the goal of selecting accounts for further data collection, experiments were conducted to understand the concerning activities exhibited by the one thousand accounts whose Twitter timelines had been collected for the analysis described in Chapter 3. Specifically, each user's bot probability and proportion of tweets containing low-credibility links were measured using the BotHunter model and low-credibility sites defined in Chapter 2.

Regression Analysis

To determine whether accounts created during bursts exhibited more concerning characteristics, the regression procedure used in Chapter 2 was applied to the features for these one thousand

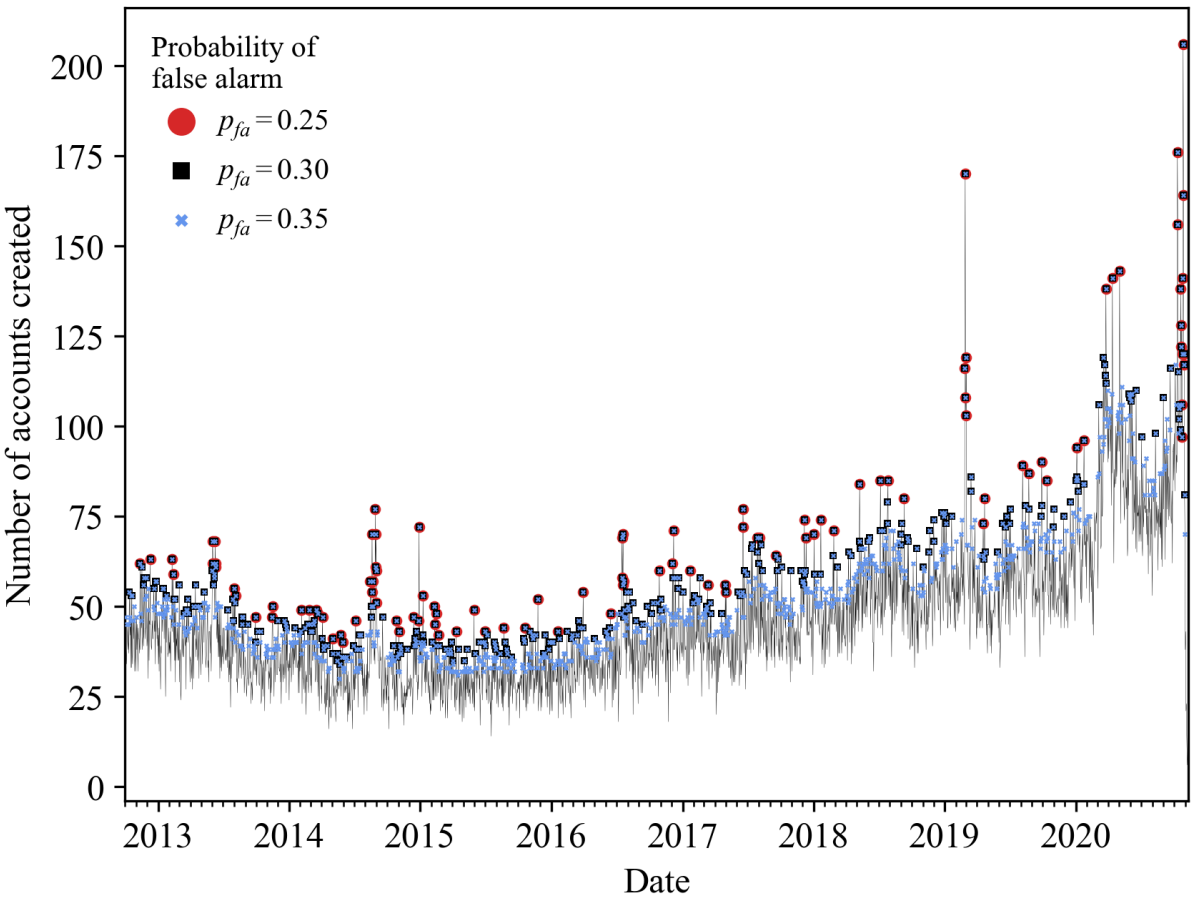


Figure 5.2: The number of accounts created each day that later appeared in the streamed sample of French terror tweets. Spikes detected using different CFAR sensitivities are indicated by different symbols.

accounts with a few modifications. First, the level of analysis became the individual account rather than each creation date. Second, the age of the account was not used as a control variable, since the goal of this experiment was not to disentangle the effects of account age and creation bursts. Rather, this experiment was designed to understand whether collecting accounts based on creation bursts could identify more concerning activity. Like in Chapter 2, the resulting p-values were adjusted to correct for a false discovery rate of five percent.

Data Collection Simulations

In addition to the regression analysis, a simple simulation was conducted to compare the impact of collecting the timelines of burst accounts to that of collecting the timelines of random accounts. In each of one thousand trials, a sample of fifty random accounts and a sample of fifty burst accounts were chosen. Each sample's mean bot probability and mean proportion of posts with low-credibility links were then calculated. For each of the two user features, the mean difference between the burst samples and the random samples was calculated and normalized by

PFA	0.25		0.30		0.35	
Dependent Variable	AME	$\frac{\text{AME}}{\sigma}$	AME	$\frac{\text{AME}}{\sigma}$	AME	$\frac{\text{AME}}{\sigma}$
Bot probability	$-0.031^{\dagger}$	-0.22	-0.035^{*}	-0.25	$-0.008^{\dagger}$	-0.06
Proportion of posts with low-credibility links	$-3\text{e}{-04}^{\dagger}$	-0.11	$2\text{e}{-04}^{\dagger}$	0.07	$8\text{e}{-05}^{\dagger}$	0.03

Table 5.3: The average marginal effects (AMEs) and associated corrected p-values for the binary burst variable from models fit to each measured user feature. Each AME was standardized by dividing by the standard deviation (σ) of the dependent variable. Complete model specifications and results are available in Appendix H. Significance: † for $p < 0.1$, * for $p < 0.05$, ** for $p < 0.01$, and *** for $p < 0.001$.

PFA	0.25		0.30		0.35	
User Feature	Δ	$\frac{\Delta}{\sigma}$	Δ	$\frac{\Delta}{\sigma}$	Δ	$\frac{\Delta}{\sigma}$
Mean bot probability	-0.0283	-0.2065	-0.0263	-0.1918	-0.0043	-0.0315
Mean proportion of posts with low-credibility links	-0.0003	-0.1045	0.0002	0.0557	$3\text{e}{-05}$	0.0091

Table 5.4: The results of the simulated data collections comparing samples of burst accounts to samples of random accounts. A positive difference Δ indicates that the feature was greater on average for the burst samples. Differences normalized by the standard deviation σ of the feature are also shown.

the standard deviation of the respective feature.

5.4.3 Results

Table 5.3 shows the average marginal effects for the independent variable indicating whether an account was created during a burst, with complete model specifications and results reported in Appendix H. The results show that in the French terror data, accounts created during bursts actually tended to be less likely to be bots, with those created during medium bursts having bot scores that were 0.25 standard deviations lower on average. In contrast, accounts created during bursts identified with a PFA of 0.30 or 0.35 exhibited slightly greater sharing of low-credibility links, but this effect was not statistically significant. This effect was reversed for the tallest spikes of account creations, though that result was also not statistically significant.

Table 5.4 shows the results of the simple data collection simulations. Unsurprisingly, these largely replicate those of the regression analysis, with the samples of burst accounts tending to have lower average bot scores. The tallest spikes still exhibited slightly lower sharing of low-credibility links among created accounts, but the medium and smaller spikes exhibited slightly greater sharing of low-credibility links.

5.4.4 Discussion

Overall, these experiments suggest that collecting user timelines based on account creation bursts would not have been effective at collecting more data from bots and would not have been particularly effective at collecting more shares of misinformation. It is likely that the difference between these results and those of Chapter 2 are due to differences in data. These experiments were conducted only on the one thousand most active users of anti-France hashtags, meaning that the data could have been biased towards authentic users. Additionally, the list of low-credibility sites was skewed towards those impacting the United States, meaning that it may have been more difficult to identify misinformation sharing in tweets discussing French issues. Finally, the original sample of French terror tweets exposed significantly fewer account creations than the data set on the 2020 US elections, so the method for identifying account creation bursts may have been less effective for the French terror tweets.

5.5 Reducing Uncertainty in Network Data

5.5.1 Background

As previously mentioned, some research has proposed using multi-stage processes for collecting better social media data [154]. Aside from using multiple stages to collect more data fitting specific qualities, this process could also be used to reduce the impact of missing information, with secondary stages dedicated to collecting data to fill in gaps identified after earlier stages.

5.5.2 Methods

Measurement of Baseline Uncertainty

To understand whether collecting more data would reduce the uncertainty associated with node centralities, several tests were conducted using the French terror tweets described in Chapter 3. First, the original streamed sample of tweets was used to form the communication network between the one thousand most active anti-France users, as described in Chapter 3. Then, the baseline uncertainties in each user's weighted degree and betweenness centralities were measured by running the unmasked RDPG bootstrap described in Chapter 4 to construct corresponding confidence intervals. The unmasked RDPG bootstrap was chosen for this experiment because it did not require making an assumption as to the amount of missing data. Additionally, social media networks are often more scale-free in nature [7, 188], and the results from Chapter 4 indicated that the RDPG bootstrap gave the best interval coverage on scale-free networks.

Iterative Simulated Data Collection

After constructing each user's sample confidence intervals, multiple trials were conducted in which more and more users' Twitter timelines were added to the streamed sample before regenerating the user communication network and re-calculating their centrality confidence intervals. In other words, at each step of the experiment, a single user was chosen for additional

data collection and that user's timeline was added to the original sample of tweets. This augmented data set was then used to form a new communication network between the users, and the unmasked RDPG bootstrap was used to construct new confidence intervals for each user's weighted degree and betweenness centralities. These steps were repeated until all one thousand users' timelines had been added to the streamed sample of tweets. The order in which the users' timelines were added to the data was determined based on either weighted degree centrality or weighted betweenness centrality. Three user orderings were tested for each: decreasing centrality, increasing centrality, and decreasing interval width. These three methods of ranking the users implemented three data collection strategies: collecting core or influential users first, collecting peripheral users first, and collecting highly uncertain users first, respectively.

Measurement of Uncertainty Change

To determine whether the uncertainty associated with a user's centrality had changed after collecting more user timelines, the width of the users' confidence intervals was compared between the original sample network and the networks augmented with user timelines. In other words, a user's original interval width was subtracted from that user's interval width for each augmented network. A negative difference would have indicated a narrower interval width after adding the user timelines, implying a reduction in uncertainty as to that user's centrality. For each augmented network, this change in uncertainty was calculated only for the users whose timelines had been collected. Additionally, several descriptive metrics were calculated to examine whether they were related to changes in the confidence interval widths, including the mean weighted degree, betweenness, authority, and hub centralities of the users whose timelines were collected. Additionally, the proportion of isolate users was calculated at each step of the simulated data collection, along with the proportion of the timeline edges that had been revealed by that point (both in terms of edge count and total weight).

5.5.3 Results

Collecting by Degree

Figures 5.3 and 5.4 show the changes in interval widths for weighted degree and betweenness centrality when collecting user timelines in decreasing order of weighted degree centrality. The results show that though collecting the first three timelines reduced interval widths for degree centrality, the vast majority of other timeline collections either increased the interval widths or left them roughly the same as in the original streamed sample. The figures also show that the first several hundred timelines tended to starkly increase confidence interval widths before a sudden drop occurred, bringing the mean change in interval widths back towards zero.

Figures 5.5 and 5.6 show the changes in interval widths for weighted degree and betweenness centrality when collecting user timelines in increasing order of weighted degree centrality. Prioritizing the timelines of peripheral users appeared to smooth out the changes in interval widths, and they seemed to get slightly smaller when collecting approximately two hundred timelines. In general, however, the timelines did not reduce uncertainty in the node centralities.

Figures 5.7 and 5.8 show the changes in interval widths for weighted degree and betweenness

centrality when collecting user timelines in decreasing order of their interval widths for degree centrality. These were the most variable results, with multiple sudden rises and drops in the changes in interval widths.

These sudden changes in interval widths suggest that the specific timeline collected can drastically impact how the confidence intervals change. Indeed, Figures 5.3 through 5.8 show that the sudden changes in interval widths were often tied to sudden changes in the mean hub score or mean authority score of the users whose timelines had been collected. Since hub and authority score are based on a spectral decomposition of the network, these changes suggest that collecting an additional timeline significantly altered the spectral properties of the network. Given that the RDPG method of bootstrapping networks is also based on the adjacency spectral embedding, significant changes in the spectral structure of a network should also change the corresponding RDPG-based confidence intervals. Additionally, the figures show that the sudden changes in interval widths were often accompanied by noticeable increases in the proportions of total edge count or network weight, implying that the most recently collected timeline had added significant information to the network. It is therefore highly plausible that the individual timeline that is collected can dramatically influence the remaining analytical uncertainty through its impact on the spectral structure of the network.

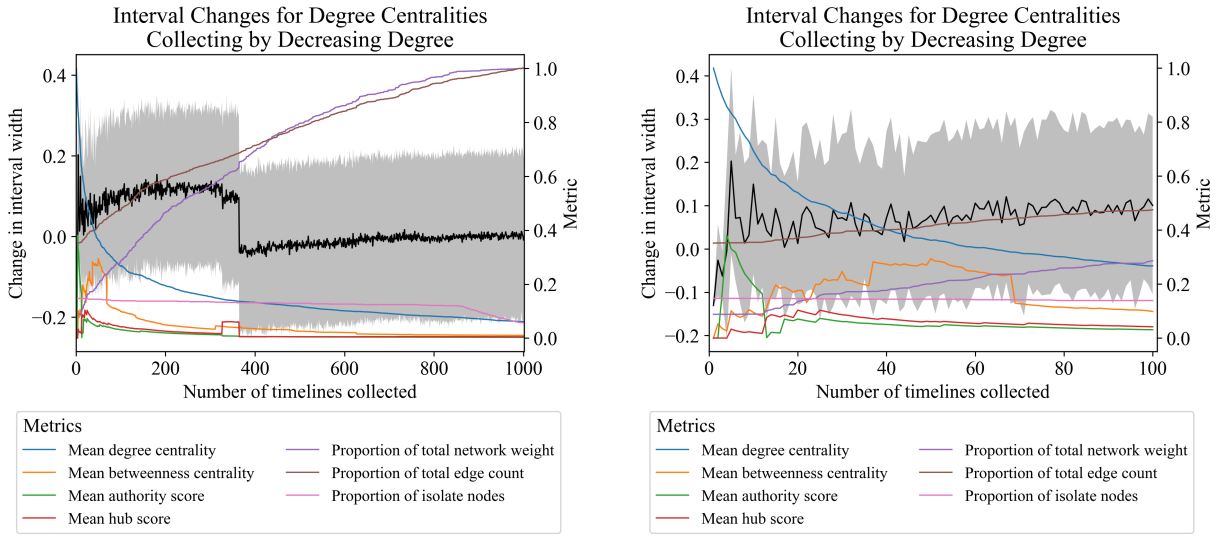
Collecting by Betweenness

Figures 5.9 and 5.10 show the changes in interval widths for weighted degree and betweenness centrality when collecting user timelines in decreasing order of weighted betweenness centrality. Both sets of results show that only the first collected user's confidence interval narrowed in width. Subsequent timeline collections tended to increase confidence interval widths on average, which matched the results from collecting based on degree. As with the degree-based results, the intervals eventually tended to return to their original widths, but this required the collection of many more timelines than when collecting based on degree.

Figures 5.11 and 5.12 show the changes in interval widths for weighted degree and betweenness centrality when collecting user timelines in increasing order of weighted betweenness centrality. Prioritizing the collection of less central users appeared to instead help narrow some confidence intervals when collecting fewer than one hundred timelines. As with the degree-based results, starting with the less central users dampened the sudden changes in interval widths.

Figures 5.13 and 5.14 show the changes in interval widths for weighted degree and betweenness centrality when collecting user timelines in decreasing order of their interval widths for betweenness centrality. Collecting the timelines of the most uncertain users only sometimes decreased interval widths on average, and these results were less dynamic than the corresponding degree-based results.

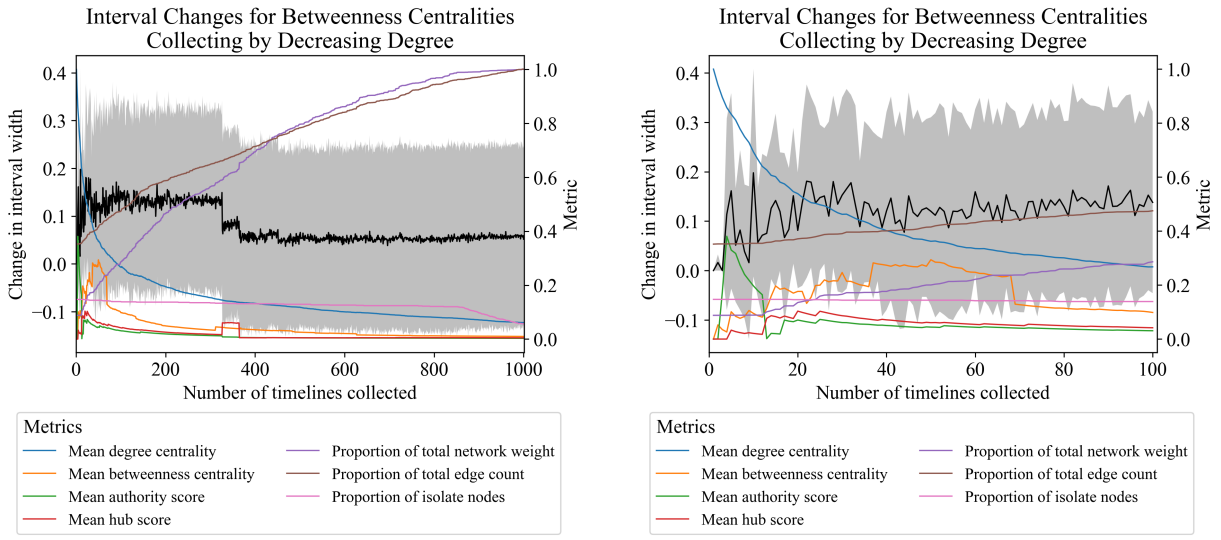
Though these changes in interval widths were often less precipitous than when ordering timeline collections based on weighted degree centrality, they were often still tied to stark changes in the mean hub or authority scores of the collected users and notable changes in the proportions of total edge count or network weight revealed by the collected timelines. The collected user's position in the network and the amount of information revealed by that user's timeline were therefore still tied to sudden changes in confidence interval widths.



(a) All timelines

(b) Focused view of first 100 timelines

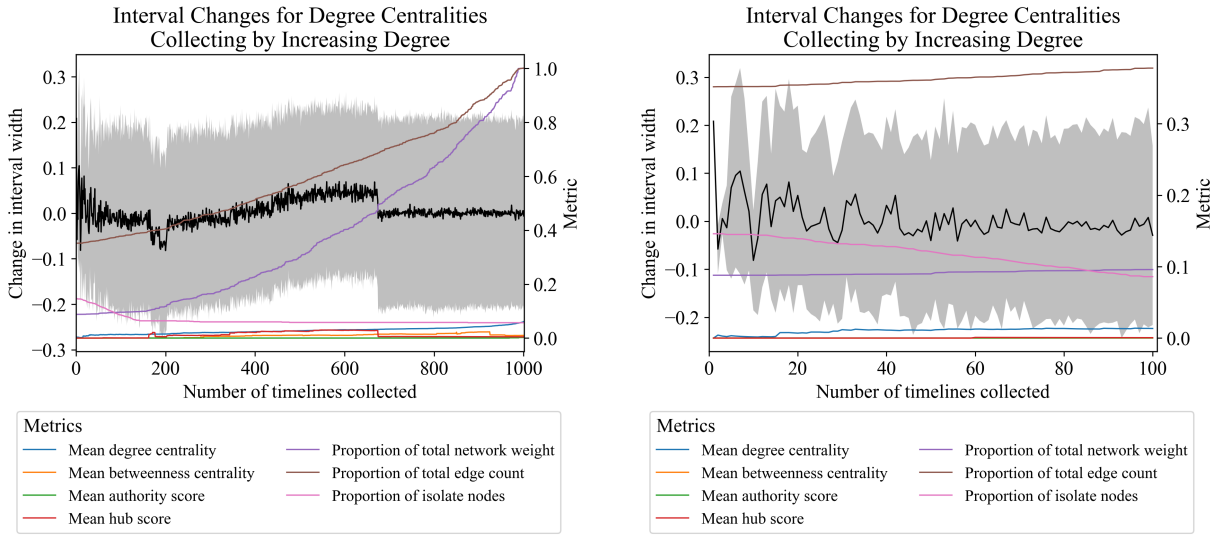
Figure 5.3: Changes in confidence interval widths for users' weighted degree centralities when adding timelines in order of decreasing weighted degree centrality in the sample communication. The black line and left axis show the mean change in width among users whose timelines were added, and the gray area indicates one standard deviation from that mean. The colored lines and right axis show metrics calculated at each collection step.



(a) All timelines

(b) Focused view of first 100 timelines

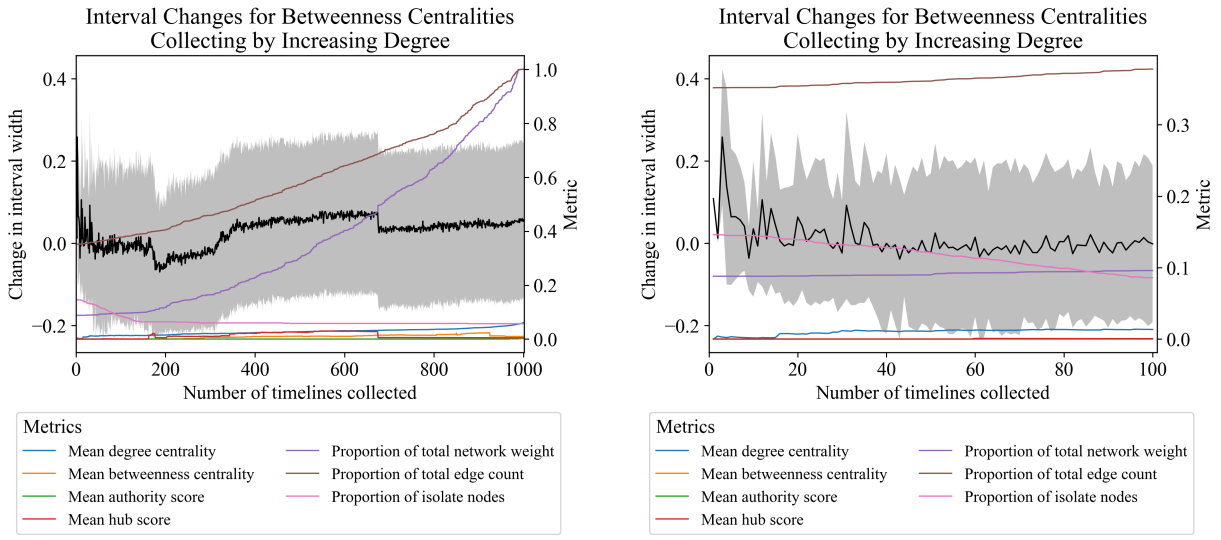
Figure 5.4: Changes in confidence interval widths for users' weighted betweenness centralities when adding timelines in order of decreasing weighted degree centrality in the sample communication. The black line and left axis show the mean change in width among users whose timelines were added, and the gray area indicates one standard deviation from that mean. The colored lines and right axis show metrics calculated at each collection step.



(a) All timelines

(b) Focused view of first 100 timelines

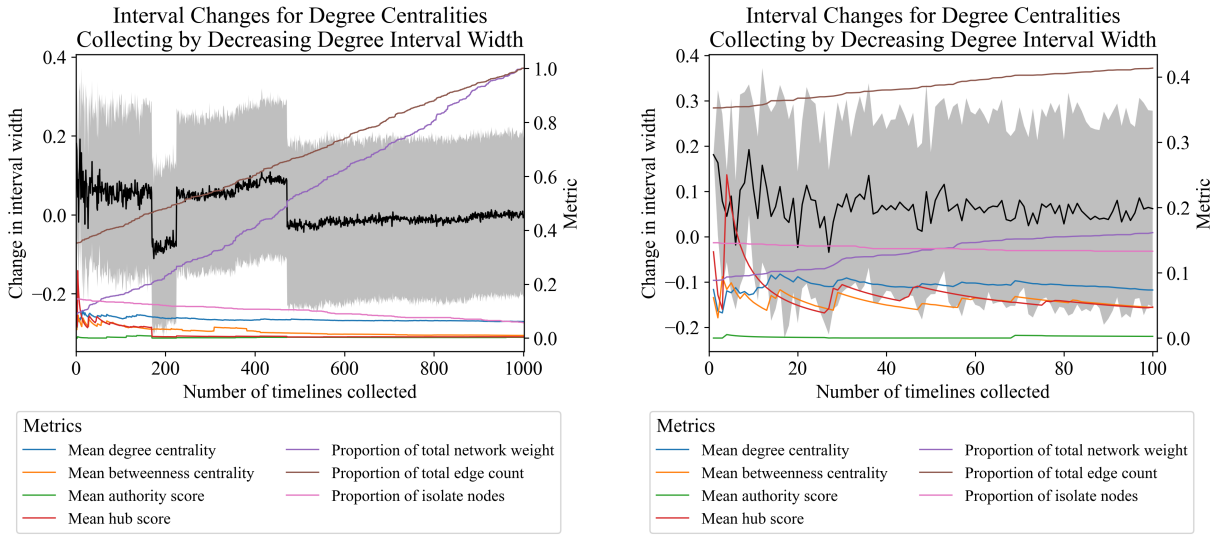
Figure 5.5: Changes in confidence interval widths for users' weighted degree centralities when adding timelines in order of increasing weighted degree centrality in the sample communication. The black line and left axis show the mean change in width among users whose timelines were added, and the gray area indicates one standard deviation from that mean. The colored lines and right axis show metrics calculated at each collection step.



(a) All timelines

(b) Focused view of first 100 timelines

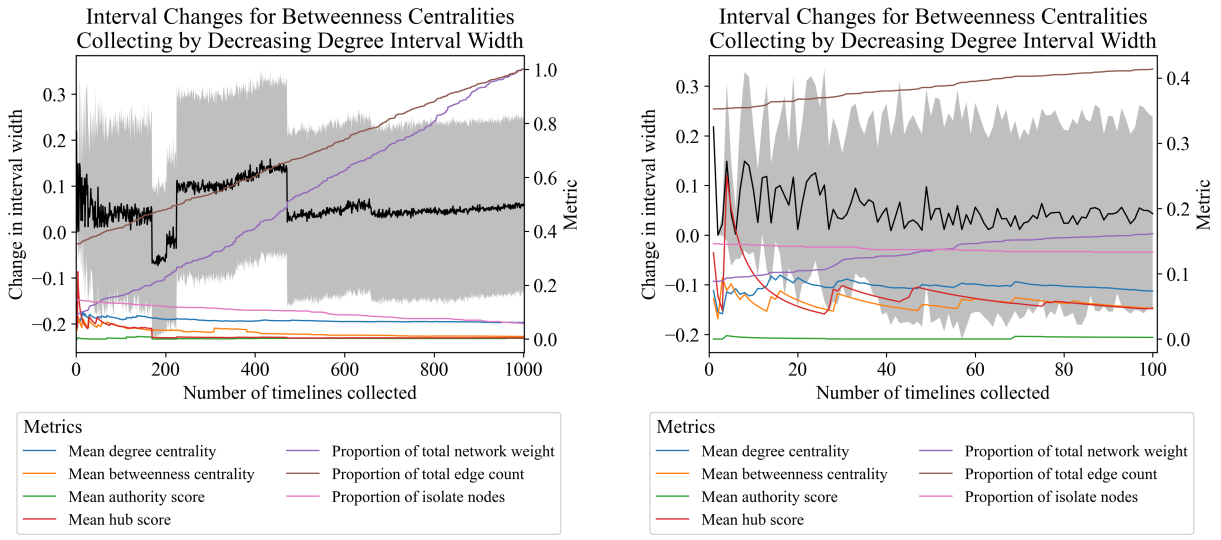
Figure 5.6: Changes in confidence interval widths for users' weighted betweenness centralities when adding user timelines in order of increasing weighted degree centrality in the sample communication. The black line and left axis show the mean change in width among users whose timelines were added, and the gray area indicates one standard deviation from that mean. The colored lines and right axis show metrics calculated at each collection step.



(a) All timelines

(b) Focused view of first 100 timelines

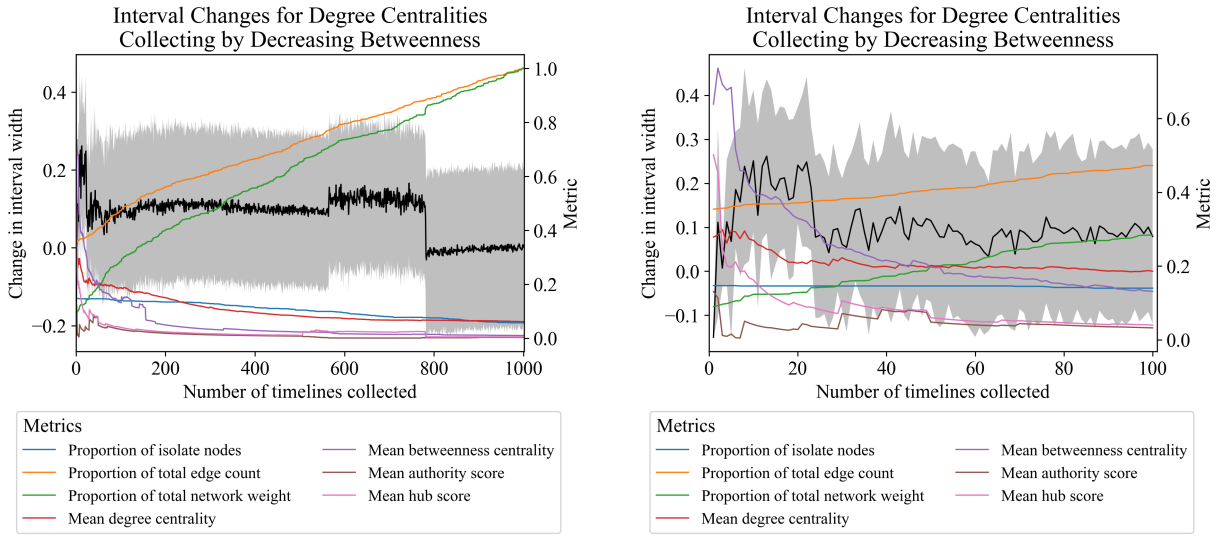
Figure 5.7: Changes in confidence interval widths for users' weighted degree centralities when adding timelines in order of decreasing interval width for weighted degree centrality in the sample communication. The black line and left axis show the mean change in width among users whose timelines were added, and the gray area indicates one standard deviation from that mean. The colored lines and right axis show metrics calculated at each collection step.



(a) All timelines

(b) Focused view of first 100 timelines

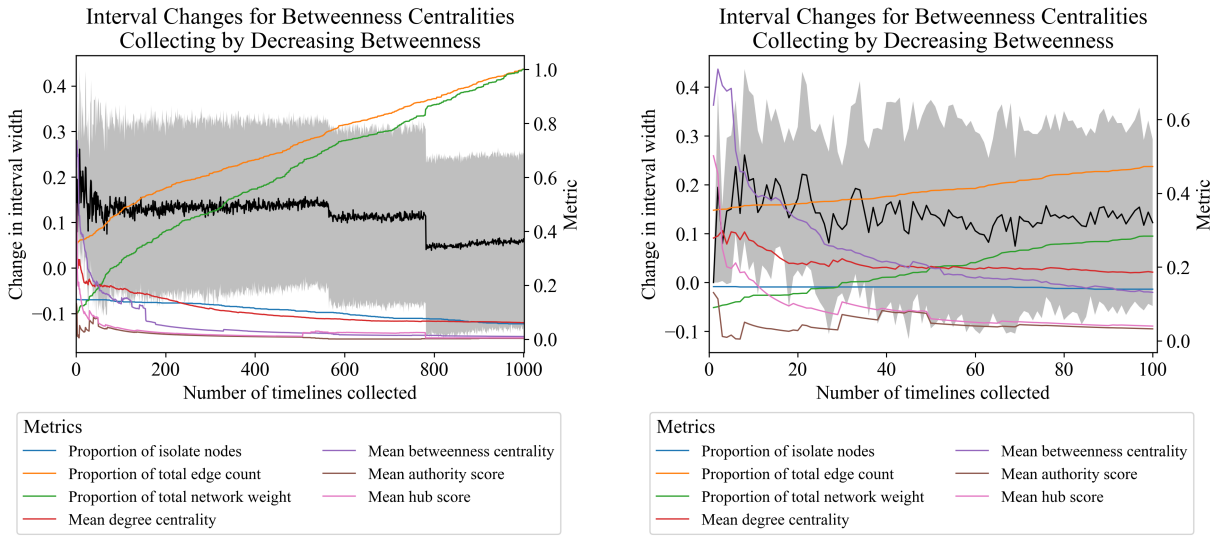
Figure 5.8: Changes in confidence interval widths for users' weighted betweenness centralities when adding timelines in order of decreasing interval width for weighted degree centrality in the sample communication. The black line and left axis show the mean change in width among users whose timelines were added, and the gray area indicates one standard deviation from that mean. The colored lines and right axis show metrics calculated at each collection step.



(a) All timelines

(b) Focused view of first 100 timelines

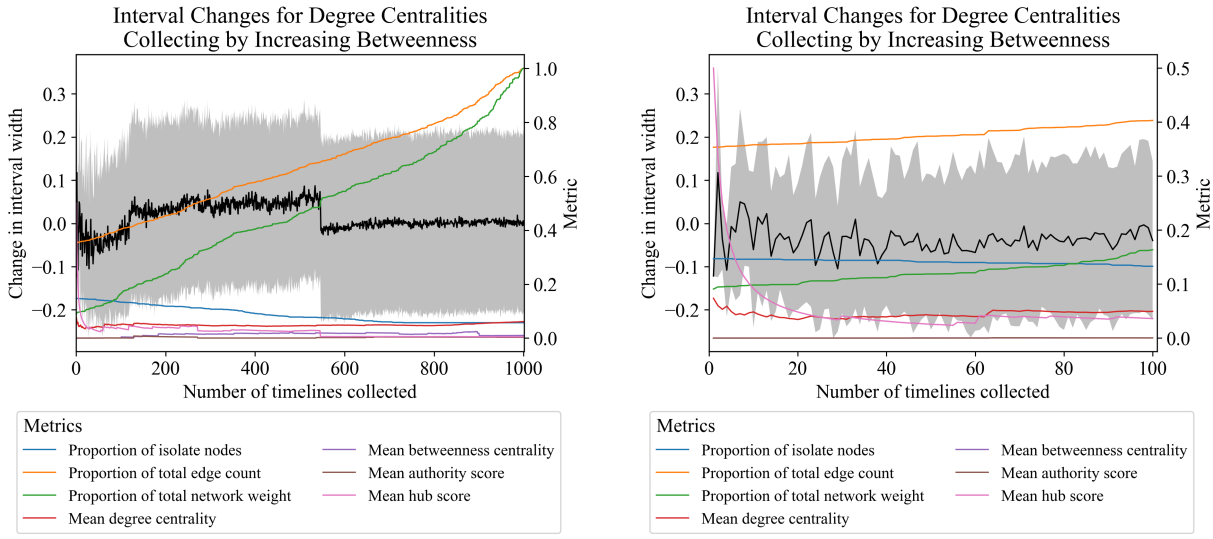
Figure 5.9: Changes in confidence interval widths for users' weighted degree centralities when adding timelines in order of decreasing weighted betweenness centrality in the sample communication. The black line and left axis show the mean change in width among users whose timelines were added, and the gray area indicates one standard deviation from that mean. The colored lines and right axis show metrics calculated at each collection step.



(a) All timelines

(b) Focused view of first 100 timelines

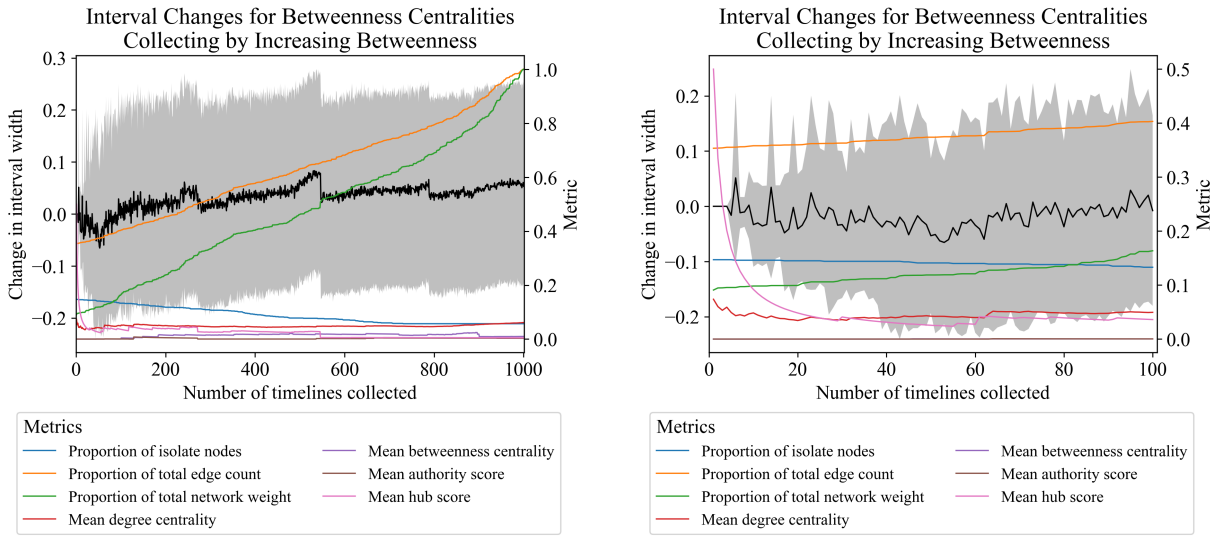
Figure 5.10: Changes in confidence interval widths for users' weighted betweenness centralities when adding timelines in order of decreasing weighted betweenness centrality in the sample communication. The black line and left axis show the mean change in width among users whose timelines were added, and the gray area indicates one standard deviation from that mean. The colored lines and right axis show metrics calculated at each collection step.



(a) All timelines

(b) Focused view of first 100 timelines

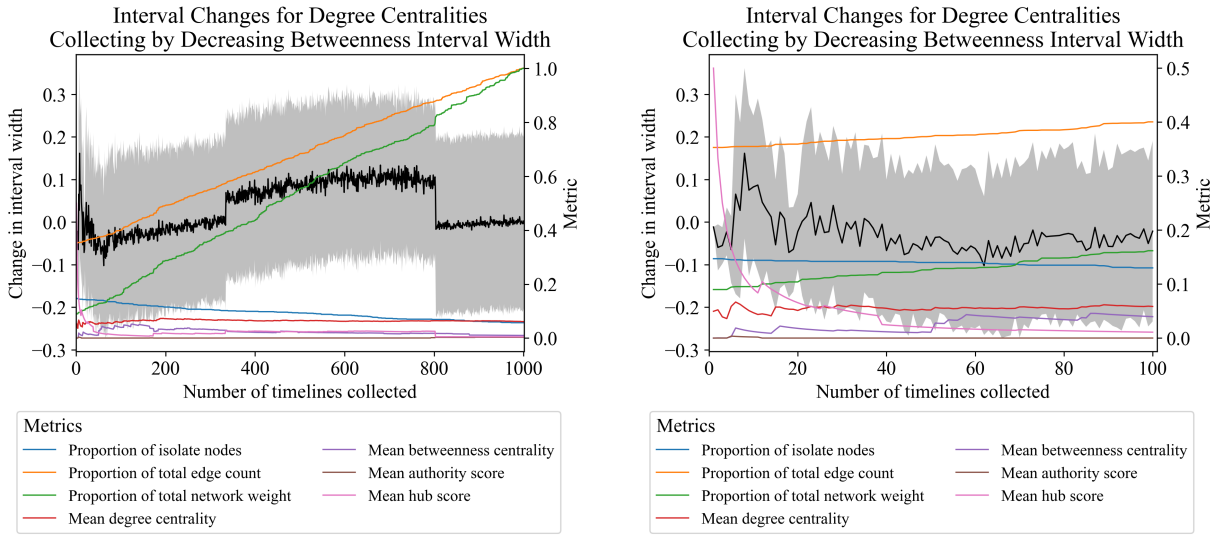
Figure 5.11: Changes in confidence interval widths for users' weighted degree centralities when adding timelines in order of increasing weighted betweenness centrality in the sample communication. The black line and left axis show the mean change in width among users whose timelines were added, and the shaded gray area indicates one standard deviation from that mean. The colored lines and right axis show metrics calculated at each collection step.



(a) All timelines

(b) Focused view of first 100 timelines

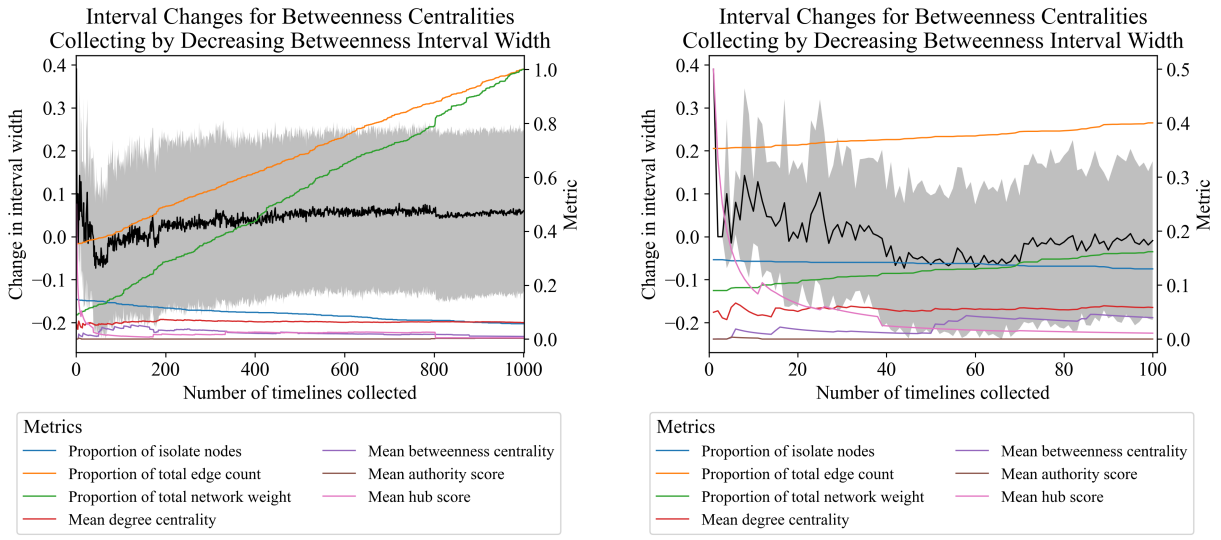
Figure 5.12: Changes in confidence interval widths for users' weighted betweenness centralities when adding timelines in order of increasing weighted betweenness centrality in the sample communication. The black line and left axis show the mean change in width among users whose timelines were added, and the shaded gray area indicates one standard deviation from that mean. The colored lines and right axis show metrics calculated at each collection step.



(a) All timelines

(b) Focused view of first 100 timelines

Figure 5.13: Changes in confidence interval widths for users' weighted degree centralities when adding timelines in order of decreasing interval width for weighted betweenness centrality in the sample communication. The black line and left axis show the mean change in width among users whose timelines were added, and the shaded gray area indicates one standard deviation from that mean. The colored lines and right axis show metrics calculated at each collection step.



(a) All timelines

(b) Focused view of first 100 timelines

Figure 5.14: Changes in confidence interval widths for users' weighted betweenness centralities when adding timelines in order of decreasing interval width for weighted betweenness centrality in the sample communication. The black line and left axis show the mean change in width among users whose timelines were added, and the gray area indicates one standard deviation from that mean. The colored lines and right axis show metrics calculated at each collection step.

5.5.4 Discussion

Overall, collecting additional data on certain users did not decrease the width of the confidence intervals associated with the centralities of users in the French terror network. Adding more data usually increased the width of the intervals, indicating there was greater uncertainty in the users' centralities. This may be a result of the weighted RDPG bootstrap, which relies on the adjacency spectral embedding (ASE) using the first singular vector and singular value. Since collecting additional timelines increased the edge weights in the user communication network, it is likely that this also inflated the network's first singular vector and singular value (when compared to the other singular vectors and values). Because the weighted RDPG bootstrap then used the (re-sampled) ASE to parameterize a set of Poisson distributions from which to draw the edge weights for each replicate network, the parameters for the Poisson distributions would also have been inflated. These increased parameters for the Poisson distributions could have been responsible for the wider confidence intervals, since increasing the parameter of a Poisson distribution also increases its variance. The greater variance in edge weights among the bootstrapped replicate networks could have been the cause for the wider intervals.

The hypothesis that changes in the network's ASE are responsible for changes in the resulting interval widths is partly supported by the results showing that changes in the average network positions of the collected users are tied to sudden changes in interval widths. The interaction between the network positions of the collected users and the functioning of the RDPG bootstrap appears to drastically impact the resulting confidence intervals, suggesting that a different method for bootstrapping the confidence intervals could have yielded very different results. However, this hypothesis for why collecting additional data increased confidence interval widths should be more rigorously tested. This could be done by conducting simulations in which the true network is generated directly from an RDPG and then sub-sampled. Simulations could also be used to test the impact of collecting additional data on the other bootstrapping methods covered in Chapter 4, and it would be interesting to see how these results vary for unweighted networks. Much more research should be done on this subject, since reducing uncertainty in network analysis is a valuable task.

5.6 Conclusion

5.6.1 Limitations

The major limitation of this chapter is that it has made use of only one data set. This means that the results for each of the four research questions are likely not applicable to other data, particularly since the French terror tweets were likely not the best data set to use for answering some of these research questions. However, applying all the techniques of this thesis to a single data set was the intention of this chapter, and it is helpful for revealing the challenges that remain in improving downstream analysis when analyzing imperfect social networks. Finally, this work did not explore the possibility of using link inference to target future data collection.

5.6.2 Contributions

This chapter applies the techniques developed in this thesis to a single data set while studying several important questions. First, this research found that link inference may be able to improve the accuracy of path-based centrality measures, whereas improving degree-based measures was more difficult. However, measures like degree and eigenvector centrality were already relatively accurate when ranking nodes in an incomplete network. This validates the notion that link inference studies should examine not just whether it improves measures of network quality [38, 91, 100] but also whether it improves downstream analysis. Second, link prediction was largely not able to increase the stability of centrality measures over time, but these measures naturally became more stable as more data were collected. This points to the development of more stable centrality measures [172] as a valuable area of research.

In terms of improving the collection of social media data, this chapter offers little useful advice. Collecting timelines for accounts created during bursts was able to identify a very small amount of additional misinformation sharing compared to collecting on random accounts, but the burst accounts were also less likely to be bots. This means that, at least in the context of these French terror discussions, tracking sudden spikes in account creations would likely not have been an effective strategy for collecting more nefarious activity. Similarly, collecting additional user timelines was not an effective strategy for narrowing the confidence intervals associated with their centralities, since in many cases adding data to the observed network actually increased interval widths and therefore made their centralities more uncertain. These results indicate that more research should be conducted on exactly how to conduct the multistage processes that some scholars [154] have advocated for collecting better social media data sets.

The principal contribution of this chapter may in fact be that it highlights how research on working with erroneous network data does not necessarily improve the downstream analysis that interests network scientists or law enforcement, like identifying central actors or nefarious activity. Future research should carefully consider how techniques developed for imperfect networks such as link inference actually impact the final results of analysis conducted on such data.

Chapter 6

Implications for Network Analysts

6.1 Contributions

By developing methods for understanding network activity from inadequate or inaccurate information, this thesis not only applies the forensic mindset [98] to social network analysis but also makes several empirical and methodological contributions to the field of network science.

6.1.1 Empirical

Chapter 2 demonstrates that sudden bursts of social media account creations might warn of future online influence campaigns. Though this idea was validated by past anecdotal reports [70, 147, 196] and manual analyses [106, 185], Chapter 2 provides statistical evidence that accounts created during bursts tended to behave more similarly, disagreed less on controversial issues, were more likely to be bots, and were more likely to share misinformation. If combined with other systems for detecting coordination [198], tracking spikes in account creations could help social media platforms pre-emptively lower the ranking of messages from suspicious accounts while monitoring their behavior. This also highlights how detecting anomalies in available information can help understand and forecast underlying network activity. Notably, the data collected for this chapter constituted the largest fully analyzed Twitter data set on the 2020 US elections.

Chapters 3 and 5 make extensive use of the set of French terror tweets. Starting from an initial sample collected as the tweets were published, the timelines for certain anti-France users were collected for this thesis. These two data sets provided opportunities for natural experiments, with the streamed sample regarded as "observed" data and the timelines interpreted as their true counterpart. By using these data to evaluate link inference algorithms in Chapter 3, this thesis adds empirical results to an area that has almost exclusively relied on simulated trials that randomly remove edges from real or stylized networks [214]. Chapter 5 also shows that for these data, it was possible to increase the accuracy of path-based centrality measures while degree-based measures were more difficult to improve. The finding that degree and eigenvector centrality were already relatively accurate at identifying top-ranking nodes is also a valuable empirical result for analysts studying samples of social media data, since analysis relying on these measures is more likely to be correct.

6.1.2 Methodological

The empirical and simulated evaluations described in Chapter 3 showed that the tensor perturbation method (TPM) was frequently more effective at identifying missing or spurious links in multidimensional networks compared to other algorithms, thereby making TPM a significant contribution to link inference for networks that are both dynamic and multiplex. The evaluation trials for identifying spurious links also presented the first known demonstration of identifying spurious connections in networks that are both dynamic and multiplex.

Next, Chapter 4 instructs network analysts on which bootstrapping methods are best for constructing confidence intervals on incomplete networks. Bootstrapping based on link inference may yield better coverage for several metrics on small-world and uniformly random networks, whereas bootstrapping based on the random dot product graph (RDPG) may be better for scale-free networks or when bounding triangle density. Chapter 4 also tests the masked version of the RDPG bootstrap [170] for constructing confidence intervals on incomplete networks. This is notable because the vast majority of prior research on constructing confidence intervals for network metrics has focused on either using those intervals to evaluate the statistical significance of observed metrics [3, 68, 136, 187] or computing intervals for a population of networks based on a perfect observation from that population [86, 123]. And most of the studies using imperfect networks to construct confidence intervals for an unseen complete network have developed methods specific to certain metrics calculated on unweighted networks [47, 162] rather than using the generic bootstrapping approach.

6.2 Significance for Practitioners

6.2.1 Network Scientists

The techniques developed in this thesis advance the analysis of imperfect networks in general. As mentioned previously, the comparison of methods for generating confidence intervals for network metrics instructs network scientists on how to best quantify the uncertainty in the results of their analysis stemming from missing edges in their data. This should make the generation of confidence intervals more straightforward for analysts across the field, particularly since reporting uncertainty is not common across network science [156].

The TPM algorithm should also allow network scientists to attain better link inference results on multidimensional networks, which could serve those who want to test the sensitivity of their analysis to missing edges by repeating it after adding inferred connections. The experiments used to evaluate TPM also highlight the factors that impact link inference performance. Network topology was extremely important, with small-world networks being much easier for link inference than the other tested topologies. Next, greater density and edge weight also appeared to increase link inference performance, with edge weight being particularly important in scale-free networks. Obviously, this performance also varied with the amount of error in the network, and Chapter 3 shows that link inference degraded more significantly when removing edges by spotlighting nodes rather than removing edges uniformly at random. Moreover, Chapter 3 underscores that network scientists studying the problem of link inference should more carefully

consider how they evaluate their algorithms, paying attention to issues related to weighted networks and the processes that cause missing links in the first place.

The results from Chapter 5 also suggest that network scientists may benefit from using link inference specifically to improve the accuracy of path-based centrality metrics calculated on incomplete networks. They may also be encouraged by the findings that weighted degree and eigenvector centrality were already relatively accurate in the tested social media network and that these centrality measures tended to stabilize on their own as additional data were collected. However, as Peel et al. [156] wrote, more work must be done to connect link inference to social network analysis more broadly. Further research on improving the accuracy and stability of node centralities would be helpful, as would research on how to improve data collection practices to reduce uncertainty in the analysis of observed networks.

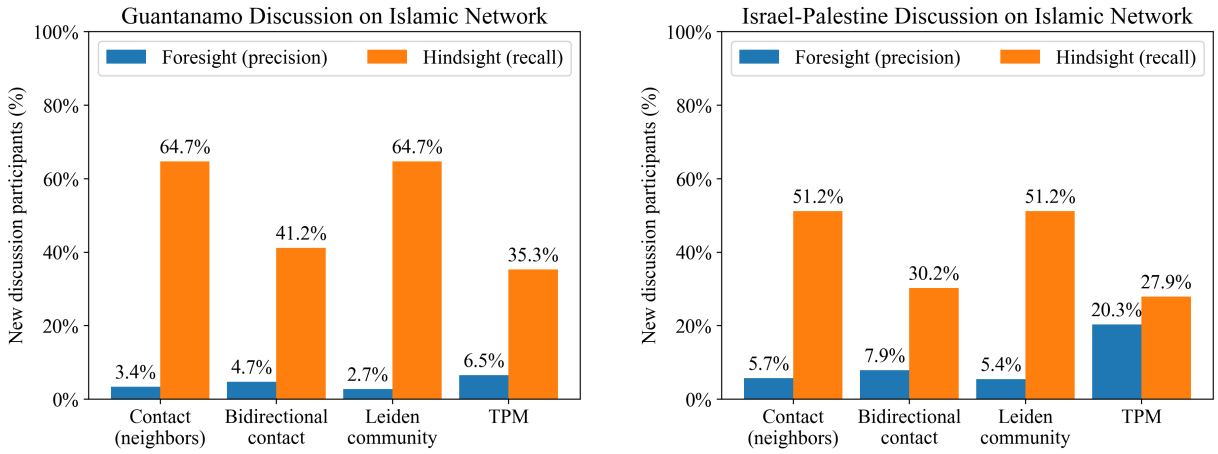
6.2.2 Law Enforcement

Aside from benefiting network scientists, this research also demonstrates how professionals in law enforcement and government intelligence can improve their analysis of covert network data. Since TPM works on dynamic multiplex networks, it can make better use of all available information while remaining flexible enough for application to diverse network data sets. This is particularly useful for law enforcement agencies, since they frequently collect information from multiple sources over time. Having better link inference capabilities should also better serve law enforcement at identifying new avenues of inquiry [33] by pointing out potential relationships to investigate.

Furthermore, using better link inference or link prediction algorithms should reduce false positives, which can be particularly problematic in law enforcement contexts. Police need to allocate their limited resources efficiently to monitor actual threats, and drawing law enforcement attention to peaceful citizens can impinge on people's civil liberties. Indeed, the automated monitoring tools used by law enforcement to identify concerning activity on social media have been criticized for potentially yielding too many false positives [124], as was partly corroborated by the analysis in Chapter 1 that tried to forecast which users would join concerning discussions on an Islamist online forum. Because of the importance of reducing false positives in this context, if these tools are to be used then they should be built upon algorithms that accomplish that goal.

To demonstrate the benefit of using more advanced link prediction algorithms, the analysis from Chapter 1 was repeated using TPM. The communication network from the Islamist online forum was separated into two un-normalized layers: one for connections from forum threads related to the particular discussion topic and the other for all other connections. The TPM algorithm was then run using the same parameters listed in Appendix C to add twenty percent more predicted connections to the layer corresponding to the topic under study. These new connections were then used to identify which users were predicted to join the discussion.

Figure 6.1 provides the results of this analysis using TPM to predict the new participants in discussions on Guantanamo Bay and the Israel-Palestine conflict. The results show that though TPM only increased the true positive rate by an additional 1.8% for the smaller Guantanamo Bay discussion, it was also able to more than double the true positive rate for the larger discussion on the Israel-Palestine conflict. Unfortunately, these increases in precision came with decreases in recall compared to other strategies, meaning that a lower proportion of the new discussion



(a) Guantanamo Bay

(b) Israel-Palestine Conflict

Figure 6.1: The success rates of various strategies for forecasting which forum users will join a discussion in the three months following its peak discussion period. Foresight (precision) indicates the proportion of users predicted by each strategy who actually ended up joining the discussion. Hindsight (recall) indicates the proportion of users joining the discussion who could have been predicted using each strategy. The contact strategy looks at who had prior contact with existing discussion participants, whereas the Leiden strategy identifies users who were in the Leiden communities [191] of any existing discussion participants. TPM refers to the tensor perturbation method described in Chapter 3.

participants was identified.

As explored by other research on the use of machine learning in policing, this precision-recall tradeoff is ultimately a policy issue [56] influenced by social priorities. If one is more interested in protecting civil rights and therefore lowering the rate of false positives, a slight increase in false negatives should be acceptable. However, when it is imperative to identify all threats, higher recall may be desired at the expense of greater false positives. For example, choosing whether or how to apply machine learning models to help maintain peace at a political protest would require policy considerations that are very different from those required when making the same decision to help protect schoolchildren from gun violence. Other tradeoffs could also exist, such as a preference to use methods with explainable meanings at the possible expense of lower effectiveness. These issues must be carefully considered within their respective contexts and communities in order to adopt the best approaches.

At a more fundamental level, the highly technical nature of the techniques for analyzing imperfect social networks calls for the involvement of digital forensics professionals, who can possess the technical backgrounds beneficial for understanding and applying the mathematical and statistical methods needed to cope with flawed network data. Despite the field of digital forensics being defined as including the analysis of digital evidence [9], digital forensics research related to social media has focused almost exclusively on the collection of data either from electronic devices or from platforms themselves [8]. Only a few scholars have called for the field of digital forensics to consider the analysis of social networks, with some calling it

social network forensics [110] or *social cyber forensics* [4]. This thesis reaffirms the need for professionals with technical backgrounds to help law enforcement apply the latest techniques for coping with imperfect data in social network analysis.

6.2.3 Social Media Analysts

As mentioned previously, the results of Chapter 2 suggest that social media platforms could do a better job of tracking spikes in account creations to place suspicious accounts under stricter scrutiny. Given that these platforms already have stated policies for lowering the ranking of concerning content in their recommendation algorithms [69, 193], they could also lower the ranking of messages produced by accounts created during bursts until the behavior of these accounts demonstrates their authenticity. This idea would ideally also be combined with other systems for detecting coordination between accounts [198], creating a more robust framework for identifying malicious influence campaigns as early as possible in order to limit their impact.

The TPM algorithm described in Chapter 3 should also yield better link inference results on social media data than other algorithms. Since social media data contain multiple kinds of interactions and content, they easily lend themselves to the construction of dynamic multiplex networks, for which existing link inference algorithms are scarce and inflexible, in that they can be computationally expensive [36, 120, 212], necessitate significant tuning [36, 150, 212], or require specific kinds of data [2, 209]. TPM also yielded better link inference performance on the French terror tweets, solidly demonstrating its effectiveness on real social media data.

Finally, Chapter 5 identifies several areas for improvement in research related to the collection of social media data. Though the results suggest that node centralities stabilize on their own as more data are collected over time, the experiments studying potential ways to target data collection processes did not provide useful guidance on how to allocate these resources more efficiently. At least in the context of the French terror tweets, collecting based on accounts created during bursts did not noticeably identify more nefarious activity, and collecting additional user timelines did not reduce the uncertainty associated with their centralities. Given that prior works have identified issues in the quality of social media data [111] or advocated for multistage data collection [154], significantly more research is needed on how to improve these collection processes while efficiently allocating limited resources.

6.3 Limitations

The methodologies used in this research impose several limitations on the presented results. The social media data used to test some of the methods likely do not generalize to all online conversations because of restrictions placed on data collection and variations between social media platforms. Additionally, different techniques for handling missing network data may work better than others depending on the network being analyzed [82], so the results of this work may not hold for all networks. This is complicated by the fact that many covert network data sets are themselves incomplete and cannot be considered ground truth, meaning that taking a sample of a network as input data and using the original as the testing set may not constitute a generalizable evaluation of a technique's effectiveness. Nevertheless, the sampling processes

used in evaluating these techniques indicate general performance trends pertinent to the kinds of networks considered during testing. Finally, the questions explored in Chapter 5 deserve much more thorough analysis using more than just a single data set, and many of them would benefit from simulation studies to exclude confounding factors.

6.4 Future Directions

This thesis points to several topics that would greatly benefit from further research. First, this thesis only examined one kind of anomaly that could be used to try to forecast future social network activity. In reality, there are many possible kinds of anomalies that can be identified in the behavior of network actors, as demonstrated by the other research in this area [96, 168, 216]. As mentioned previously, research on detecting coordination is particularly necessary for identifying online influence campaigns [55].

Second, there is still opportunity to improve the performance of TPM. As mentioned in Chapter 3, domain expertise could be exploited by modifying the noise model used in the algorithm or by encoding node attribute information in the provided network tensor. Additionally, since past work has already shown that different link inference algorithms work best on different kinds of data and that ensemble methods are more effective [82], research could be done to explore how TPM can be combined with other algorithms to further improve link inference on dynamic multiplex networks. This may even involve learning how to split a network into constituent parts that are better served by one algorithm over another.

Third, this thesis joins Peel et al. [156] in advocating for significantly more research on how to effectively use link inference and link prediction to improve subsequent social network analysis as opposed to studying it solely as an independent task. Whether that means increasing the accuracy of node centralities, improving community detection methods, or bettering other kinds of analysis, more work should be dedicated to measuring the impact of link inference on the final results concluded by social network analysts. Furthermore, link inference and link prediction could be used to improve data collection processes in the first place, thereby improving future analysis. Inferring a connection between two individuals could guide analysts to further investigate that relationship by re-interviewing study participants, putting individuals under additional surveillance, targeting their social media data collection, or otherwise adapting their data acquisition process. Future research could, for example, use the French terror tweets to test whether it is effective to collect the timelines of the users participating in the greatest number of inferred connections.

Fourth, the notion of constructing confidence intervals for metrics calculated on imperfect networks deserves much more attention, both in terms of developing better bootstrapping methods and studying their effectiveness in practice, particularly on real data. Narrowing the widths of these confidence intervals would be a valuable goal. Moreover, the question of how to narrow such intervals by collecting more information should be studied in much greater depth using simulations and additional data sets. The evaluation of different confidence interval methods could also be extended to networks with spurious edges, which were not considered in this thesis's testing of different network bootstrapping approaches.

Finally, an important remaining issue in the analysis of imperfect networks is that of estimat-

ing the amount of error in the observed data. Most link inference algorithms generate likelihood scores for edges, so ultimately one must decide how many edges to deem as missing or spurious. Similarly, several of the network bootstrapping approaches must be provided with the amount of missing edges. Research developing a method for estimating the degree of error in a social network would therefore be extremely helpful for applying the techniques developed in this thesis to the forensic analysis of observed social networks.

Appendix A

Data Collection & Processing for 2020 US Elections[1]

A.1 Terms Tracked in Data Collection

From 1 December 2019 to 17 February 2021, the Twitter streaming API was used to collect tweets relevant to the 2020 US elections. To remain relevant to election-related discussions, the specific terms being tracked were changed over the course of the collection period. These terms, hashtags, and accounts are listed below.

1 December 2019 to 16 August 2020

#election2020, #2020_presidential_election, #maga2020, #flipitblue, #keepitblue, #yeswecan, #yang2020, #JoeBiden, #BernieSanders, #ElizabethWarren, #PeteButtigieg, #FeelTheBern, #democrats, #republicans, #Bloomberg2020, #Booker, @TulsiGabbard, @GovBillWeld, @AndrewYang, @TomSteyer, @JohnDelaney, @ewarren, @WalshFreedom, @PeteButtigieg, @BernieSanders, @Devalpatrick, @MichaelBennet, @AmyKlobuchar, @TulsiGabbard, @marwilliamson, @JulianCastro, @MikeBloomberg, @CoryBooker, @JoeBiden, @realDonaldTrump

16 August 2020 to 15 January 2021

#USPS, #VoteByMail, #SaveTheUSPS, #voterfraud, #BlackLivesMatter, #BLM, #reopen, #reopenamerica, #IranSanctions, #QAnon, #WWG1WGA, “natural born”, @USPS, #election2020, #presidentialelection, #democrats, #republicans, #JoeBiden, #BidenHarris2020, #Biden, #MAGA, #KAG, @JoeBiden, @realDonaldTrump, @POTUS, @Mike_Pence, @VP, @KamalaHarris, @SenKamalaHarris, @CoryGardner, @SenCoryGardner, @Hickenlooper, @Perduesenate, @sendavidperdue, @ossoff, @joniernst, @SenJoniErnst, @GreenfieldIowa, @SenSusanCollins, @SenatorCollins, @SaraGideon, @SteveDaines, @stevebullockmt, @GovernorBullock, @ThomTillis, @SenThomTillis, @CalforNC

[1]Parts of this appendix were published in the *Journal of Big Data* [13]. The article is available on request.

15 January 2021 to 17 February 2021

@JoeBiden, @realDonaldTrump, @POTUS, @Mike_Pence, @VP, @KamalaHarris, @SenKamalaHarris, @USPS, inauguration, #Inauguration, #InaugurationDay, #Capitol, #USCapitol, #USCapital, #NationalMall, #Jan20, #election2020, #presidentialelection, #democrats, #republicans, #JoeBiden, #Biden, #MAGA, #KAG, #Trump, #USPS, #VoteByMail, #SaveTheUSPS, #voterfraud, #BlackLivesMatter, #BLM, #reopen, #reopenamerica, #IranSanctions, #QAnon, #WWG1WGA, "natural born"

A.2 Hashtags Used to Initialize Stance Detection

Determining user stance required seeding the algorithm with initial sets of labeled hashtags. After sorting hashtags by popularity, the following hashtags were manually labeled as being in favor of or against mail-in voting and mask wearing.

Hashtags in Favor of Mail-in Voting

#savetheusps, #votebymail, #votebymail2020, #saveusps, #uspsisessential, #dontmesswithusps, #votebymailearly, #mailthevotega, #protectuspswhistleblowers, #usps2020vote, #protectusps, #votebymail4bernie, #usps_is_essential, #fundusps, #wewillvotebymail, #mailthevotefl, #protecttheusps, #votebluebymail2020, #savetheusps2020, #supportusps, #votebymailnow, #fundtheusps, #uspsismylifeline, #saveourusps, #uspsforever, #voteabsenteemailblue, #dontmesswiththeusps, #votebluebymail, #restoreusps, #votebymailforall

Hashtags Against Mail-in Voting

#mailinvoterfraud, #mailinballotfraud, #nomailinvoting, #exposeusps, #nomailinballots, #novotebymail, #mailinvotingfraud, #demvotebymailscam, #uspslies, #mailvoterfraud, #hellnotovotebymail, #mailinvotefraud, #stopvotebymail, #ballotbymailfraud, #mailinballotsfraud, #defundusps, #nomailinvote, #votebymailfraud, #defundtheusps, #votebymailisfraud, #exposetheusps, #nomailinvotes, #fraudulentmailinballots, #mailinballotswouldbeatotaldisater, #stopmailinvoting, #fakemailvotes, #nomailvoting, #dontmailyourballot, #uspssucks, #voterfraudbymail

Hashtags in Favor of Mask Wearing

#wearamask, #wearadamnmask, #maskup, #maskssavelives, #wearamasksavealife, #masks4all, #realmenwearmasks, #maskupamerica, #wearthedamnmask, #masksforall, #wearthemask, #mandatemasks, #wearyourmask, #makefacemaskmandatory, #wearamaskplease

Hashtags Against Mask Wearing

#masksoffamerica, #masksoff, #unmaskamerica, #masksdontwork, #nomaskmandates, #nomoremasks, #maskoff, #nomaskonme, #burnyourmask, #masksdonotwork, #takeoffthemask,

#maskhoax, #burnthemask, #sheepwearmasks, #trashthemask

A.3 Low-Credibility Web Domains

Users were deemed to have shared a link to a low-credibility site if the link led to any of the following Web domains.

100percentfedup.com, 10news.one, 12minutos.com, 16wmpo.com, 20minutenews.com, 247newsmedia.com, 24aktuelles.com, 24online.news, 24usainfo.com, 24wpn.com, 24x365live.com, 911truth.org, a-news24.com, aattp.org, abcbusinessnews.com, abcnews-us.com, abcnews.com.co, abovetopsecret.com, acpeds.org, actionnews3.com, activistmommy.com, activistpost.com, actualite.co, actualites.co, adflegal.org, adobochronicles.com, afa.net, ageofautism.com, ageofshitlords.com, alabamaobserver.com, aldipest.com, alertchild.com, allenwestrepublic.com, allnewspipeline.com, alphanewsmn.com, alternativenews.com, althealthworks.com, altright.com, alynews.com, amazingnews.net, americafans.com, americanflare.com, americanflavor.news, americanfreepress.net, americanjournalreview.com, americanlookout.com, americannews.com, americanpatriotdaily.com, americanpeoplenetwork.com, americanpoliticnews.co, americanpresident.co, americanprides.com, americanprides.org, americanprinciplesproject.org, americansfortruth.com, americanthinker.com, americantoday.news, americantoday.us, americantruthseekers.com, americanupdater.com, americasfreedomfighters.com, americasnewest.com, americatalks.com, amposts.com, amren.com, amtvmedia.com, anews-24.com, anews24.org, angrypatriotmovement.com, anonhq.com, anonjekloy.tk, answersingenesis.org, antinews.com, antinewsnetwork.com, areyouasleep.com, armyusanews.com, asamericanasapplepie.org, asheepnomore.net, assassinationscience.com, associatedmediacoverage.com, aurora-news.us, australiannationalreview.com, autisminvestigated.com, avoiceformen.com, awarenessact.com, backroombuzz.com, badcriminals.com, baldwinpost.com, baltimoregazette.com, barenakedislam.com, battypost.com, bb4sp.com, beaware.one, beforeitsnews.com, bients.com, bigbluevision.org, biggovernment.news, bigleaguepolitics.com, bignuggetnews.com, billionbibles.org, bipartisanreport.com, bizstandardnews.com, blackeyepolitics.com, blackgenocide.org, blackpigeonspeaks.com, blingnews.com, bloodandsoil.org, blueinformer.com, bluetribune.com, bluevision.news, bluevisionpost.com, bluntforcetruth.com, borderherald.com, borderlandalternativemedia.com, bostonleader.com, breaking911.com, breakingnews247.net, breakingnews365.net, breakingnewsblast.com, breakingtop.world, breitbart.com, britainfirst.tv, burrardstreetjournal.com, butthatsnoneofmybusiness.com, buzzfeedusa.com, canadafreepress.com, cap-news.com, carm.org, cartelpress.com, cartelreport.com, cbsnews.com.co, cei.org, celebtricity.com, cfact.org, channel16news.com, channel17news.com, channel18news.com, channel22news.com, channel23news.com, channel24news.com, channel28news.com, channel34news.com, channel45news.com, channel5000.com, checkpointasia.net, chicagodaily.pro, choiceandtruth.com, christianaction.org, christianscience.com, christiantimesnewspaper.com, christiantoday.info, cityworldnews.com, civictribune.com, clancyreport.com, clarionproject.org, clashdaily.com, clear-politics.com, climatechangedispatch.com, climatescienceinternational.org, climatism.blog,

cloverchronicle.com, cnewsgo.com, cnn-business-news.ga, cnn-globalnews.com, cnnews3.com, cnsnews.com, coasttocoastam.com, coffeebreakforyou.com, collective-evolution.com, collectivelyconscious.net, com.de, concernedwomen.org, consciouslyenlightened.com, conservative101.com, conservativearmy88.com, conservativebuzz.com, conservativebyte.com, conservativedailypost.com, conservativefighters.com, conservativeflashnews.com, conservativefreepress.com, conservativeinfocorner.com, conservativeinfocorner.us, conservativeinsider.co, conservativemedia.com, conservativepaper.com, conservativepoliticus.com, conservativepost.com, conservativespirit.com, conservativestudio.com, conservativeview.info, consmovement.com, consnation.com, conspatriot.com, conspiracydailyupdate.com, conspiracyinquirer.com, conspiracyplanet.com, cooltobeconservative.com, corbettreport.com, cosmicintelligenceagency.com, countdowntozerotime.com, creation.com, csglobe.com, curiousmindmagazine.com, daily-sun.com, dailybuzzlive.com, dailycurrant.com, dailyfeed.news, dailyfinesser.com, dailyheadlines.net, dailyhealthpost.com, dailyinfobox.com, dailyinsidernews.com, dailymail.co.uk, dailynews10.com, dailynews11.com, dailynews3.com, dailynews33.com, dailynews5.com, dailynewsposts.info, dailynotify.com, dailyoccupation.com, dailypresser.com, dailysidnews.com, dailystormer.name, dailysurge.com, dailythings.world, dailyusaupdate.com, dailyworldinformation.com, damnleaks.com, damreports.com, dangerandplay.com, darkjournalist.com, dataasylum.com, davidicke.com, dcclothesline.com, dcdirtylaundry.com, debunkingskeptics.com, defendevropa.org, defenseusa.club, defiantamerica.com, defund.com, democraticmoms.com, democraticreview.com, democraticunderground.com, denverguardian.com, denverinquirer.com, departed.co, departedme.com, departedmedia.com, deplorablekel.com, deplorablemovement2020.news, digifection.com, dineal.com, disclose.tv, discoverthenetworks.org, discovery.org, diyhours.net, donaldtrumpnews.co, donaldtrumppotus45.com, downtrend.com, drainingtheswamp.info, drugsofficial.com, eaglerising.com, earthpulse.com, educate-yourself.org, embols.com, empireherald.com, empirenews.net, empiresports.co, en-volve.com, endingthefed.com, endthefed.org, endtimeheadlines.org, enhlive.com, escapeallthesethings.com, everydaybreakingnews.com, everynewshere.com, evolutionnews.org, ewao.com, explainlife.com, exposinggovernment.com, fairus.org, famousviralstories.com, fanzinger.com, fbnewscycle.com, federalistnation.com, federalisttribune.com, fedsalert.com, fellowshipoftheminds.com, flashinfo.org, flashnewscorner.com, floridasunpost.com, focusnews.info, focusonthefamily.com, folksvideo.com, foreverconscious.com, forfreedomworld.com, fox-news24.com, france24-tv.com, frc.org, freakoutnation.com, freddymag.com, freedomadvocates.org, freedomcrossroads.us, freedomdaily.com, freedomlibertynews.com, freedomproject.com, freedomsfinalstand.com, freeinfomedia.com, freepatriotpost.com, freshdailyreport.com, fromthetrenchesworldreport.com, frontpagemag.com, fury.news, gellerreport.com, geoengineeringwatch.org, getcancerremedies.com, girlsjustwannahaveguns.com, givemeliberty01.com, globalassociatednews.com, globalinfotoday.com, globalpoliticsnow.com, globalresearch.ca, globalrevolutionnetwork.com, globalskywatch.com, globemagazine.com, godandscience.org, godlikeproductions.com, goneleft.com, goodgopher.com, gotnews.com, gotquestions.org, govtslaves.com, greatamericanrepublic.com, greenvillegazette.com, guerilla.news, guerillanews.com, gulagbound.com, gummypost.com, haarp.net, halturnershow.com, hangthebankers.com, hawarnews.com, headlinebrief.com,

healingoracle.ch, healthimpactnews.com, healthnutnews.com, healthy-holistic-living.com, healthy-vibes.com, healthyworldhouse.com, heartland.org, heaviermetal.net, hoggwatch.com, holylandnutrition.com, hotglobalnews.com, houstonchronicle-tv.com, hsionline.com, humansarefree.com, huzlers.com, iceagenow.info, icr.org, identityevropa.com, ifyouonlynews.com, ihatethemedia.com, ihavethetruth.com, ihealthtube.com, ihr.org, illicitinfo.com, illuminati-news.com, illuminatiwatcher.com, ilovemyfreedom.org, imowired.com, indigenous.club, infiniteunknown.net, infogalactic.com, informationliberation.com, informedowners.com, infowars.com, intellihub.com, interestingdailynews.com, intrendtoday.com, iowaclimate.org, ipatriot.com, isthatlegit.com, itaglive.com, jesus-is-savior.com, jesusdaily.com, jewsnews.co.il, jewwatch.com, joeforamerica.com, jokerviral.com, jonesreport.com, jookos.com, judicialwatch.org, junkscience.com, kata33.com, kbc14.com, kcna.kp, kcst7.com, kf13.com, klponews.com, km8news.com, kmt11.com, knightstemplarinternational.com, knightstemplarorder.com, knowledgeoftoday.org, knp7.com, konkonsagh.biz, krb7.com, krbcnews.com, kspm33.com, kty24news.com, kupr7.com, ky12news.com, ky6news.com, kypo6.com, ladbible.com, ladylibertysnews.com, landrypost.com, larouchepac.com, lastdeplorables.com, lastresistance.com, learnprogress.org, learntherisk.org, leftexposed.org, lewrockwell.com, liberalplug.com, liberalsociety.com, liberty-courier.com, libertyalliance.com, libertybrief.com, libertyisviral.com, libertyvideos.org, libertywritersnews.com, local31news.com, localnews33.com, londonwebnews.com, loonwatch.com, lopezreport.com, madamericannetwork.com, madworldnews.com, maga2020.us, magavoter.com, mainerepublicemailalert.com, majorthoughts.com, martinlutherking.org, maywoodpost.com, mbganews.com, mbynews.com, mckenziepost.com, mediamass.net, mediazone.news, medicine.news, meforum.org, mercola.com, metapedia.org, metropolitanworlds.com, mississippiherald.com, mixi.media, molonlabemedia.com, moonbattery.com, morningledger.com, morningnewsusa.com, movingleft.com, mrcblog.com, mrnewswatch.com, msfanpage.link, msnbc.website, mygoogle.ca, mynewsguru.com, myrightamerica.com, natall.com, nationaleconomicseditorial.com, nationalenquirer.com, nationalinsiderpolitics.com, nationalpolicy.institute, nationalreport.net, nationalvanguard.org, nativestuff.us, naturalcuresnotmedicine.info, naturalmedicine.news, naturalnews.com, naturalnewsblogs.com, nbc.com.co, ncscooper.com, needtoknow.news, neonnettle.com, nephef.com, neverrepublican.com, nevo.news, newcenturytimes.com, newnation.org, newobserveronline.com, newpoliticstoday.com, news14kgpn.com, news14now.com, news4ktla.com, newsandstoriesfromusa.com, newsbiscuit.com, newsbreakhere.com, newsbreakingspipe.com, newsbreakshere.com, newsbuzzdaily.com, newsbysquad.com, newschicken.com, newsconservative.com, newsdaily10.com, newsdaily12.com, newsdaily27.com, newsexaminer.net, newsfeedhunter.com, newsfeedobserver.com, newsformetoday.com, newshubs.info, newsinsideout.com, newsleak.co, newslo.com, newsner.com, newsnow17.com, newsoftrump.com, newsphd.com, newspunch.com, newstarget.com, newsuptoday.com, newswars.com, newswatch33.com, newswithviews.com, newzmagazine.com, nmws.us, nnettle.com, nodisinfo.com, noscomunicamos.com, notallowedto.com, noticias365.info, notrickszone.com, now8news.com, nowtheendbegins.com, npiamerica.org, nvic.org, nydailynews-tv.com, nymeta.co, oathkeepers.org, observeronline.news, occupydemocrats.com,

onepoliticalplaza.com, onlineconservativepress.com, openmagazines.com, oreillypost.com, other98.com, oureyeonislam.com, ourhealthguides.com, ourlandofthefree.com, oye.news, pacificpundit.com, pakalertpress.com, pamelageller.com, patribotics.blog, patriotbeacon.com, patriotcrier.com, patriotfires.com, patriothangout.com, patrioticexpress.com, patrioticviralnews.com, patriotsvoice.info, patriotupdate.com, patriotusa.website, persecutes.com, personalliberty.com, plymouthpatriotpost.com, politicalblindspot.com, politicalears.com, politicalflare.com, politicalinbox.com, politicalmayhem.news, politicalsitenews.com, politicoinfo.com, politicops.com, politicot.com, politics.com, politicslive.net, politicspaper.com, politicsusanews.com, politifact.news, powderedwigsociety.com, powerdaily.us, powerofnative.com, prageru.com, prepareforchange.net, president45donaldtrump.com, pressunion.org, prideof-america.org, principia-scientific.org, prisonplanet.com, prntly.com, proamericanews.com, profam.org, progressivestoday.com, projectpurge.com, prophecynewswatch.com, prophecytoday.com, proud-patriots.com, proudcons.com, proudleader.com, puppetstringnews.com, qanon.pub, qualitysharing.com, rapturenewsnetwork.com, react365.com, readconservatives.news, realconservativesunite.com, realfarmacy.com, realjewnews.com, realnews24.com, realnewsrightnow.com, realtimepolitics.com, rearfront.com, redcountry.us, redice.tv, redinfo.us, redpeople.us, redpolitics.us, redrocktribune.com, redstatejournalist.com, redstatepundit.com, redstatewatcher.com, redstatewave.com, redwhiteandright.com, reedcooper.net, reflectionofmind.org, religionmind.com, remnantnewspaper.com, renewamerica.com, rense.com, researchantisemitism.ca, returnofkings.com, returntonow.net, revolutionradio.org, revolutions2040.com, rickwells.us, rightalerts.com, rightjournalist.com, rightsidenews.com, rightwing.news, rightwingnews.com, rilenews.com, rinf.com, rodong.rep.kp, rogue-nation3.com, rrtribune.com, rt.com, rumorjournal.com, ruthinstitute.org, rwnofficial.com, sanevax.org, satiratribune.com, savethemales.ca, scrapetv.com, sgtreport.com, sharednewsreport.com, shariawatch.org.uk, sheepkillers.com, shoebat.com, shtfplan.com, sickchirpse.com, skeptiko.com, smhwtfnews.com, smobserved.com, smoloko.com, snoopack.com, socialeverythings.com, southernconservativeextra.com, southjerseymechanic.com, speakgov.com, speisa.com, spinzon.com, sputniknews.com, squawker.org, staresattheworld.com, statefort.com, statenation.co, stateofthenation.co, stateofthenation2012.com, states-tv.com, stgeorgegazette.com, stonecoldtruth.com, stormcloudsgathering.com, stormfront.org, straightstoned.com, strategic-culture.org, stuppid.com, subjectpolitics.com, success-street.com, summit.news, sundayinquirer.com, supremepatriot.com, surenews.com, survivalblog.com, sustainablepulse.com, svijmedia.com, swampdrain.com, takimag.com, tashnews.com, tdnewswire.com, tdtalliance.com, teaparty.org, teddystick.com, telegraphsun.com, teoinfo.com, tfp.org, the-global-news.com, the-insider.co, the-postillon.com, theamericanmirror.com, theavocadonews.com, theaware.net, thebostontribune.com, thebreakingnews.co, thebreakingnews.today, thecommonsenseshow.com, thecontroversialfiles.net, thedailybell.com, thedailyconspiracy.com, thedailysheeple.com, thedcgazette.com, thedeplorablesociety.com, theduran.com, theeventchronicle.com, theexaminer.site, thefederalistpapers.org, theforbiddenknowledge.com, thefreepatriot.org, thefreethoughtproject.com, thefrt.com, thegatewaypundit.com, thegoldwater.com, thegopwatchdog.com, thehealthconsciousness.com, thehornnews.com, theinternationalreporter.org, theinternetpost.net, thelastgreatstand.com, thelastlineofdefense.org,

thelibertybeacon.com, themiamigazette.com, themillenniumreport.com, themindunleashed.com, themoralofthestory.us, thenationalpatriot.com, thenationalsun.com, thenet24h.com, thenewsnerd.com, thenewyorkevening.com, thenochill.com, thepeoplestruth.com, thepoliticalinsider.com, thepoliticaltribune.com, thepolitics.online, thepremiumnews.com, thepublicdiscourse.com, theracketreport.com, theredelephants.com, thereligionofpeace.com, thereporterz.com, therightists.com, therightstuff.biz, therundownlive.com, theseattletribune.com, thetrumpmedia.com, thetruthdivision.com, thetruthseeker.co.uk, theusa-news.com, theusaconservative.com, theuspatriot.com, thevalleyreport.com, theviralpatriot.com, thewashingtonpress.com, thewashingtonpundit.com, theworldupdate.com, thinkamericana.com, thinkingmomsrevolution.com, thirdestatenewsgroup.com, thoughtcrimeradio.net, thrivemovement.com, times.com.mx, timespolice.com, tmn.today, tmzbreaking.com, tmzbusiness.com, tmzcomedy.com, tmzhiphop.com, tmzuncut.com, tmzurban.com, tmzworldnews.com, tmzworldstarnews.com, todaydispatch.com, topinfopost.com, toprightnews.com, toptopic.club, tpusa.com, triggerreset.net, trueactivist.com, trueamericans.me, truebluescoop.com, truepundit.com, truetrumpers.com, trueviralnews.com, trump.news, trumpers.online, trumpservativenews.info, trunews.com, truthandaction.org, truthbroadcastnetwork.com, truthchannelpolitics.com, truthcommand.com, truthexam.com, truthfeednews.com, truthinmedia.com, truthkings.com, truthmonitor.com, truthrevolt.org, truthuncensored.net, turningpoint.news, twisted.news, unclesamsmisguidedchildren.com, undergroundhealth.com, undergroundjournalist.org, undergroundnewsreport.com, understandingthethreat.com, uniquewebmagazine.com, unitedmediapublishing.com, universepolitics.com, unz.com, us24news.com, usa-conservative.com, usa-radio.com, usa-television.com, usa360-tv.com, usaconservativereport.com, usadailyinfo.com, usadailypost.us, usadailyreview.com, usadailythings24.com, usadailytime.com, usadosenews.com, usafirstinformation.com, usafortrumponline.com, usahitman.com, usainfront.com, usanetwork.info, usanews4u.us, usanewsflash.com, usanewshome.com, usanewstoday.com, usanewstoday.org, usapolitics24hrs.com, usapoliticsnow.com, usapoliticstoday.com, usapoliticszone.com, usapubliclife.com, usareally.com, usasnich.com, usasupreme.com, usatodaynews.me, usatodaypolitics.com, usaviralnewstoday.com, usawatchdog.com, usaworldbox.com, uschronicle.com, ushealthyadvisor.com, ushealthylife.com, usherald.com, usinfonews.com, usjournalreview.com, uslibertywire.club, uslibertywire.com, uspoliticsinfo.com, uspoln.com, uspostman.com, ustruthwire.com, vaccines.news, vaccinesrevealed.com, vactruth.com, vaxxter.com, vdare.com, vesselnews.io, veteransfordonaldtrump.com, vidmax.com, vigilantcitizen.com, viralactions.com, viralcocaine.com, viralcords.com, viraldevil.com, viralliberty.com, viralmugshot.com, viralnewsnetwork.net, viralpropaganda.com, viralspeech.com, viralstuppid.com, voiceofamericatv.com, voiceofeurope.com, voxtribune.com, vrevealed.com, wakeup-world.com, wakingtimes.com, walkwithher.com, wallbuilders.com, washingtonevening.com, washingtonfeed.com, washingtonpost.com.co, wattsupwiththat.com, wazanews.tk, wcpm3.com, webdaily.com, weconservative.com, weekendherald.com, weekendpoliticalnews.com, werk35.com, westernsentinel.com, westfieldpost.com, wetheproudpatriots.com, wftj8news.com, whale.to, whatdoesitmean.com, whatreallyhappened.com, whatsupic.com, whydontyoutrythis.com, winningdemocrats.com, witscience.org, wleb21.com, wm21news.com, wmacnews.com, wmb36.com, wnd.com, world-politicus.com, worldaffairsbrief.com,

worldinformation24.info, worldnewscircle.com, worldnewsdailyreport.com, worldnewspolitics.com, worldpoliticsnow.com, worldpoliticus.com, worldtruth.tv, wrejnews.com, wrpm33.com, wrpt16.com, wtoe5news.com, wy21news.com, x22report.com, xbn-news.com, yesimright.com, yiannopoulos.net, yournewswire.com, zootfeed.com

Appendix B

Regression Results for Account Creation Bursts in 2020 US Election Tweets[1]

B.1 Communication With Others

Tables B.1 through B.3 contain the complete specifications for the regression models fit to predict daily average neighborhood similarities, whereas Tables B.4 through B.6 contain information on the corresponding average marginal effects. Tables B.7 through B.9 contain the complete specifications for the regression models fit to predict daily average clustering coefficients, whereas Tables B.10 through B.12 contain information on the corresponding average marginal effects.

B.2 Talk Amongst Contemporaries

Tables B.13 through B.15 contain the complete specifications for the regression models fit to predict the density of communication between accounts created on each day, whereas Tables B.16 through B.18 contain information on the corresponding average marginal effects.

B.3 Hashtag Use

Tables B.19 through B.21 contain the complete specifications for the regression models fit to predict the average hashtag similarities between accounts created on each day, whereas Tables B.22 through B.24 contain information on the corresponding average marginal effects.

B.4 Agreement on Political Issues

Tables B.25 through B.27 contain the complete specifications for the regression models fit to predict agreement on mail-in voting among accounts created on each day, whereas Tables B.28 through B.30 contain information on the corresponding average marginal effects. Tables B.31

[1]Parts of this appendix were published in the *Journal of Big Data* [13]. The article is available on request.

through B.33 contain the complete specifications for the regression models fit to predict agreement on mask wearing among accounts created on each day, whereas Tables B.34 through B.36 contain information on the corresponding average marginal effects.

B.5 Hashtag Synchrony

Tables B.37 through B.39 contain the complete specifications for the regression models fit to predict the average proportion of account pairs that synchronized their hashtag usages among accounts created on each day, whereas Tables B.40 through B.42 contain information on the corresponding average marginal effects.

B.6 Bot Activity

Tables B.43 through B.45 contain the complete specifications for the regression models fit to predict the daily proportions of created accounts with bot scores greater than or equal to 70%, whereas Tables B.46 through B.48 contain information on the corresponding average marginal effects.

B.7 Sharing of Low-Credibility Sources

Tables B.49 through B.51 contain the complete specifications for the regression models fit to predict the daily proportions of created accounts who shared at least one link to a low-credibility site, whereas Tables B.52 through B.54 contain information on the corresponding average marginal effects.

Mean Neighborhood Cosine Similarity, PFA = 0.25

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-0.939623	0.009106	-103.191966	$0e{+}00$	
Burst	0.241688	0.052452	4.607789	0.000004	0.000010
Age	-0.017155	0.001341	-12.791957	$2e{-}37$	$1e{-}36$
N	3654				
Fit RMSD	0.033950				
$\frac{\text{RMSD}}{\sigma}$	0.944295				

Table B.1: Specification for a one-part fractional regression model fit to predict the mean neighborhood cosine similarity among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (*N*), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Neighborhood Cosine Similarity, PFA = 0.30

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-0.944542	0.008949	-105.543472	$0e{+}00$	
Burst	0.197605	0.033653	5.871783	$4e{-}09$	$2e{-}08$
Age	-0.016725	0.001322	-12.653784	$1e{-}36$	$7e{-}36$
N	3654				
Fit RMSD	0.033749				
$\frac{\text{RMSD}}{\sigma}$	0.938708				

Table B.2: Specification for a one-part fractional regression model fit to predict the mean neighborhood cosine similarity among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (*N*), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Neighborhood Cosine Similarity, PFA = 0.35

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-0.947639	0.009050	-104.708522	$0e{+}00$	
Burst	0.084748	0.009007	9.409533	$5e{-}21$	$3e{-}20$
Age	-0.017996	0.001345	-13.381445	$8e{-}41$	$6e{-}40$
N	3654				
Fit RMSD	0.033829				
$\frac{\text{RMSD}}{\sigma}$	0.940918				

Table B.3: Specification for a one-part fractional regression model fit to predict the mean neighborhood cosine similarity among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Neighborhood Cosine Similarity, PFA = 0.25

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.049528	0.011279	4.391350	0.000011	0.000025	1.377587
Age	-0.003337	0.000264	-12.628227	$1e{-}36$	$9e{-}36$	-0.092823
N	3654					
Fit RMSD	0.033950					
$\frac{\text{RMSD}}{\sigma}$	0.944295					

Table B.4: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the mean neighborhood cosine similarity among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Neighborhood Cosine Similarity, PFA = 0.30

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.040075	0.007105	5.640436	2e−08	6e−08	1.114664
Age	−0.003253	0.000260	−12.495337	8e−36	5e−35	−0.090486
N	3654					
Fit RMSD	0.033749					
$\frac{\text{RMSD}}{\sigma}$	0.938708					

Table B.5: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the mean neighborhood cosine similarity among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Neighborhood Cosine Similarity, PFA = 0.35

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.016693	0.001801	9.270715	2e−20	1e−19	0.464317
Age	−0.003501	0.000265	−13.200758	9e−40	7e−39	−0.097367
N	3654					
Fit RMSD	0.033829					
$\frac{\text{RMSD}}{\sigma}$	0.940918					

Table B.6: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the mean neighborhood cosine similarity among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Clustering Coefficient, PFA = 0.25

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-0.380297	0.002135	-178.127880	$0e{+}00$	
Burst	0.046281	0.017987	2.573000	0.010082	0.014916
Age	-0.013557	0.000316	-42.898436	$0e{+}00$	$0e{+}00$
N	3654				
Fit RMSD	0.010998				
$\frac{\text{RMSD}}{\sigma}$	0.754211				

Table B.7: Specification for a one-part fractional regression model fit to predict the mean clustering coefficient among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Clustering Coefficient, PFA = 0.30

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-0.380751	0.002123	-179.312128	$0e{+}00$	
Burst	0.031065	0.009776	3.177553	0.001485	0.002587
Age	-0.013527	0.000315	-42.893349	$0e{+}00$	$0e{+}00$
N	3654				
Fit RMSD	0.011001				
$\frac{\text{RMSD}}{\sigma}$	0.754395				

Table B.8: Specification for a one-part fractional regression model fit to predict the mean clustering coefficient among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Clustering Coefficient, PFA = 0.35

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-0.379359	0.002173	-174.577943	$0e{+}00$	
Burst	0.003714	0.002466	1.506123	0.132036	0.158589
Age	-0.013750	0.000331	-41.581123	$0e{+}00$	$0e{+}00$
N	3654				
Fit RMSD	0.011069				
$\frac{\text{RMSD}}{\sigma}$	0.759110				

Table B.9: Specification for a one-part fractional regression model fit to predict the mean clustering coefficient among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Clustering Coefficient, PFA = 0.25

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.011059	0.004318	2.560883	0.010441	0.015035	0.758412
Age	-0.003224	0.000075	-42.728435	$0e{+}00$	$0e{+}00$	-0.221092
N	3654					
Fit RMSD	0.010998					
$\frac{\text{RMSD}}{\sigma}$	0.754211					

Table B.10: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the mean clustering coefficient among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Clustering Coefficient, PFA = 0.30

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.007411	0.002340	3.167381	0.001538	0.002637	0.508216
Age	−0.003217	0.000075	−42.723493	0e+00	0e+00	−0.220609
N	3654					
Fit RMSD	0.011001					
$\frac{\text{RMSD}}{\sigma}$	0.754395					

Table B.11: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the mean clustering coefficient among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Clustering Coefficient, PFA = 0.35

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.000884	0.000587	1.505647	0.132158	0.158589	0.060591
Age	−0.003270	0.000079	−41.409442	0e+00	0e+00	−0.224250
N	3654					
Fit RMSD	0.011069					
$\frac{\text{RMSD}}{\sigma}$	0.759110					

Table B.12: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the mean clustering coefficient among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Communication Density, PFA = 0.25

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-13.467652	0.055090	-244.467578	$0e{+}00$	
Burst	0.747510	0.181863	4.110282	0.000040	0.000079
Age	-0.041269	0.008267	-4.992077	$6e{-}07$	0.000002
N	3654				
Fit RMSD	0.000002				
$\frac{\text{RMSD}}{\sigma}$	0.993878				

Table B.13: Specification for a one-part fractional regression model fit to predict the communication density among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Communication Density, PFA = 0.30

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-13.479059	0.056545	-238.375697	$0e{+}00$	
Burst	0.565231	0.120409	4.694262	0.000003	0.000007
Age	-0.040507	0.008404	-4.820098	0.000001	0.000004
N	3654				
Fit RMSD	0.000002				
$\frac{\text{RMSD}}{\sigma}$	0.994084				

Table B.14: Specification for a one-part fractional regression model fit to predict the communication density among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Communication Density, PFA = 0.35

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-13.483481	0.059789	-225.517495	$0e{+}00$	
Burst	0.253917	0.057777	4.394755	0.000011	0.000025
Age	-0.045362	0.008421	-5.386634	$7e{-}08$	$3e{-}07$
N	3654				
Fit RMSD	0.000002				
$\frac{\text{RMSD}}{\sigma}$	0.995120				

Table B.15: Specification for a one-part fractional regression model fit to predict the communication density among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Communication Density, PFA = 0.25

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.000001	$4e{-}07$	2.933526	0.003351	0.005568	0.617066
Age	$-5e{-}08$	$1e{-}08$	-4.744361	0.000002	0.000005	-0.023302
N	3654					
Fit RMSD	0.000002					
$\frac{\text{RMSD}}{\sigma}$	0.993878					

Table B.16: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the communication density among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Communication Density, PFA = 0.30

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	9e−07	2e−07	3.682793	0.000231	0.000437	0.418457
Age	−5e−08	1e−08	−4.583396	0.000005	0.000011	−0.022872
N	3654					
Fit RMSD	0.000002					
$\frac{\text{RMSD}}{\sigma}$	0.994084					

Table B.17: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the communication density among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Communication Density, PFA = 0.35

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	3e−07	8e−08	4.211855	0.000025	0.000053	0.154952
Age	−5e−08	1e−08	−5.088630	4e−07	0.000001	−0.025613
N	3654					
Fit RMSD	0.000002					
$\frac{\text{RMSD}}{\sigma}$	0.995120					

Table B.18: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the communication density among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Hashtag Cosine Similarity, PFA = 0.25

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-5.044978	0.023656	-213.261896	$0e{+}00$	
Burst	0.928195	0.299209	3.102161	0.001921	0.003242
Age	0.004245	0.003986	1.064983	0.286884	0.316157
N	3654				
Fit RMSD	0.004534				
$\frac{\text{RMSD}}{\sigma}$	0.970439				

Table B.19: Specification for a one-part fractional regression model fit to predict the mean hashtag cosine similarity among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Hashtag Cosine Similarity, PFA = 0.30

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-5.038551	0.026311	-191.496349	$0e{+}00$	
Burst	0.532626	0.201521	2.643030	0.008217	0.012478
Age	0.002886	0.004585	0.629455	0.529051	0.560172
N	3654				
Fit RMSD	0.004608				
$\frac{\text{RMSD}}{\sigma}$	0.986266				

Table B.20: Specification for a one-part fractional regression model fit to predict the mean hashtag cosine similarity among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Hashtag Cosine Similarity, PFA = 0.35

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-5.019228	0.034176	-146.865852	$0e{+}00$	
Burst	0.130817	0.053937	2.425339	0.015294	0.021734
Age	-0.001799	0.006578	-0.273395	0.784550	0.800011
N	3654				
Fit RMSD	0.004658				
$\frac{\text{RMSD}}{\sigma}$	0.996975				

Table B.21: Specification for a one-part fractional regression model fit to predict the mean hashtag cosine similarity among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Hashtag Cosine Similarity, PFA = 0.25

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.009838	0.004819	2.041599	0.041191	0.052960	2.105514
Age	0.000028	0.000026	1.076154	0.281858	0.313822	0.006022
N	3654					
Fit RMSD	0.004534					
$\frac{\text{RMSD}}{\sigma}$	0.970439					

Table B.22: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the mean hashtag cosine similarity among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Hashtag Cosine Similarity, PFA = 0.30

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.004546	0.002208	2.058501	0.039542	0.051452	0.972833
Age	0.000019	0.000030	0.633691	0.526282	0.560172	0.004094
N	3654					
Fit RMSD	0.004608					
$\frac{\text{RMSD}}{\sigma}$	0.986266					

Table B.23: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the mean hashtag cosine similarity among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Mean Hashtag Cosine Similarity, PFA = 0.35

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.000903	0.000397	2.275951	0.022849	0.031074	0.193214
Age	−0.000012	0.000044	−0.272554	0.785196	0.800011	−0.002551
N	3654					
Fit RMSD	0.004658					
$\frac{\text{RMSD}}{\sigma}$	0.996975					

Table B.24: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the mean hashtag cosine similarity among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Standard Deviation of Mail-In Voting Stances, PFA = 0.25

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-0.933589	0.001417	-658.732848	$0e{+}00$	
Burst	-0.082691	0.019757	-4.185366	0.000028	0.000058
Age	0.000819	0.000221	3.695522	0.000219	0.000423
N	3654				
Fit RMSD	0.006127				
$\frac{\text{RMSD}}{\sigma}$	0.948729				

Table B.25: Specification for a one-part fractional regression model fit to predict the standard deviation of mail-in voting stances among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (*N*), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Standard Deviation of Mail-In Voting Stances, PFA = 0.30

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-0.933435	0.001466	-636.692188	$0e{+}00$	
Burst	-0.045515	0.009444	-4.819523	0.000001	0.000004
Age	0.000837	0.000231	3.626332	0.000287	0.000526
N	3654				
Fit RMSD	0.006227				
$\frac{\text{RMSD}}{\sigma}$	0.964288				

Table B.26: Specification for a one-part fractional regression model fit to predict the standard deviation of mail-in voting stances among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (*N*), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Standard Deviation of Mail-In Voting Stances, PFA = 0.35

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-0.934328	0.001564	-597.305554	$0e{+}00$	
Burst	-0.010969	0.002050	-5.350109	$9e{-}08$	$3e{-}07$
Age	0.001140	0.000264	4.315924	0.000016	0.000034
N	3654				
Fit RMSD	0.006363				
$\frac{\text{RMSD}}{\sigma}$	0.985239				

Table B.27: Specification for a one-part fractional regression model fit to predict the standard deviation of mail-in voting stances among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (*N*), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Standard Deviation of Mail-In Voting Stances, PFA = 0.25

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	-0.016475	0.003863	-4.265149	0.000020	0.000042	-2.551159
Age	0.000166	0.000045	3.697498	0.000218	0.000423	0.025707
N	3654					
Fit RMSD	0.006127					
$\frac{\text{RMSD}}{\sigma}$	0.948729					

Table B.28: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the standard deviation of mail-in voting stances among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Standard Deviation of Mail-In Voting Stances, PFA = 0.30

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	-0.009145	0.001878	-4.868486	0.000001	0.000003	-1.416130
Age	0.000170	0.000047	3.628395	0.000285	0.000526	0.026297
N	3654					
Fit RMSD	0.006227					
$\frac{\text{RMSD}}{\sigma}$	0.964288					

Table B.29: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the standard deviation of mail-in voting stances among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Standard Deviation of Mail-In Voting Stances, PFA = 0.35

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	-0.002221	0.000414	-5.361753	$8\text{e}{-08}$	$3\text{e}{-07}$	-0.343998
Age	0.000231	0.000054	4.319160	0.000016	0.000034	0.035814
N	3654					
Fit RMSD	0.006363					
$\frac{\text{RMSD}}{\sigma}$	0.985239					

Table B.30: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the standard deviation of mail-in voting stances among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Standard Deviation of Mask Wearing Stances, PFA = 0.25

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-1.021737	0.001473	-693.452742	$0e{+}00$	
Burst	-0.056760	0.020746	-2.735946	0.006220	0.010026
Age	-0.002047	0.000228	-8.967588	$3e{-}19$	$2e{-}18$
N	3654				
Fit RMSD	0.005662				
$\frac{\text{RMSD}}{\sigma}$	0.961260				

Table B.31: Specification for a one-part fractional regression model fit to predict the standard deviation of mask wearing stances among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (*N*), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Standard Deviation of Mask Wearing Stances, PFA = 0.30

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-1.022026	0.001528	-668.665448	$0e{+}00$	
Burst	-0.025465	0.009650	-2.638864	0.008318	0.012478
Age	-0.001990	0.000239	-8.335742	$8e{-}17$	$4e{-}16$
N	3654				
Fit RMSD	0.005741				
$\frac{\text{RMSD}}{\sigma}$	0.974743				

Table B.32: Specification for a one-part fractional regression model fit to predict the standard deviation of mask wearing stances among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (*N*), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Standard Deviation of Mask Wearing Stances, PFA = 0.35

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-1.022670	0.001612	-634.460486	$0e{+}00$	
Burst	-0.005450	0.002001	-2.724299	0.006444	0.010086
Age	-0.001818	0.000270	-6.722029	$2e{-}11$	$7e{-}11$
N	3654				
Fit RMSD	0.005788				
$\frac{\text{RMSD}}{\sigma}$	0.982712				

Table B.33: Specification for a one-part fractional regression model fit to predict the standard deviation of mask wearing stances among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Standard Deviation of Mask Wearing Stances, PFA = 0.25

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	-0.010845	0.003909	-2.774219	0.005533	0.009055	-1.841324
Age	-0.000396	0.000044	-8.952917	$3e{-}19$	$2e{-}18$	-0.067290
N	3654					
Fit RMSD	0.005662					
$\frac{\text{RMSD}}{\sigma}$	0.961260					

Table B.34: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the standard deviation of mask wearing stances among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Standard Deviation of Mask Wearing Stances, PFA = 0.30

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	-0.004902	0.001846	-2.655001	0.007931	0.012236	-0.832364
Age	-0.000385	0.000046	-8.322314	$9\text{e}{-17}$	$4\text{e}{-16}$	-0.065429
N	3654					
Fit RMSD	0.005741					
$\frac{\text{RMSD}}{\sigma}$	0.974743					

Table B.35: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the standard deviation of mask wearing stances among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Standard Deviation of Mask Wearing Stances, PFA = 0.35

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	-0.001054	0.000387	-2.727530	0.006381	0.010086	-0.179022
Age	-0.000352	0.000052	-6.712755	$2\text{e}{-11}$	$7\text{e}{-11}$	-0.059749
N	3654					
Fit RMSD	0.005788					
$\frac{\text{RMSD}}{\sigma}$	0.982712					

Table B.36: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the standard deviation of mask wearing stances among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Account Pairs Synchronizing Hashtags, PFA = 0.25

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-7.468893	0.187519	-39.830104	$0e{+}00$	
Burst	0.579887	0.297436	1.949620	0.051221	0.065081
Age	-0.070728	0.030109	-2.349065	0.018821	0.026398
N	3651				
Fit RMSD	0.001488				
$\frac{\text{RMSD}}{\sigma}$	0.997273				

Table B.37: Specification for a one-part fractional regression model fit to predict the proportion of account pairs synchronizing hashtags among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (*N*), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Account Pairs Synchronizing Hashtags, PFA = 0.30

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-7.474022	0.190818	-39.168237	$0e{+}00$	
Burst	0.401222	0.210180	1.908943	0.056269	0.070664
Age	-0.070526	0.030386	-2.321012	0.020286	0.028089
N	3651				
Fit RMSD	0.001488				
$\frac{\text{RMSD}}{\sigma}$	0.997357				

Table B.38: Specification for a one-part fractional regression model fit to predict the proportion of account pairs synchronizing hashtags among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (*N*), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Account Pairs Synchronizing Hashtags, PFA = 0.35

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-7.502358	0.150812	-49.746578	$0e{+}00$	
Burst	0.277604	0.200411	1.385176	0.165999	0.197009
Age	-0.073261	0.028581	-2.563257	0.010370	0.015035
N	3651				
Fit RMSD	0.001488				
$\frac{\text{RMSD}}{\sigma}$	0.996836				

Table B.39: Specification for a one-part fractional regression model fit to predict the proportion of account pairs synchronizing hashtags among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (*N*), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Account Pairs Synchronizing Hashtags, PFA = 0.25

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.000321	0.000209	1.532119	0.125493	0.154014	0.214995
Age	-0.000029	0.000014	-2.102493	0.035510	0.047347	-0.019611
N	3651					
Fit RMSD	0.001488					
$\frac{\text{RMSD}}{\sigma}$	0.997273					

Table B.40: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the proportion of account pairs synchronizing hashtags among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Account Pairs Synchronizing Hashtags, PFA = 0.30

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.000201	0.000121	1.655499	0.097823	0.121436	0.134503
Age	−0.000029	0.000014	−2.079499	0.037572	0.049484	−0.019555
N	3651					
Fit RMSD	0.001488					
$\frac{\text{RMSD}}{\sigma}$	0.997357					

Table B.41: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the proportion of account pairs synchronizing hashtags among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Account Pairs Synchronizing Hashtags, PFA = 0.35

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.000125	0.000102	1.224918	0.220606	0.253462	0.083736
Age	−0.000030	0.000013	−2.273143	0.023018	0.031074	−0.020313
N	3651					
Fit RMSD	0.001488					
$\frac{\text{RMSD}}{\sigma}$	0.996836					

Table B.42: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the proportion of account pairs synchronizing hashtags among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Bot Accounts, PFA = 0.25

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-1.198462	0.011793	-101.623192	$0e{+}00$	
Burst	0.161144	0.032017	5.033061	$5e{-}07$	0.000001
Age	-0.111096	0.001812	-61.317141	$0e{+}00$	$0e{+}00$
N	3654				
Fit RMSD	0.031500				
$\frac{\text{RMSD}}{\sigma}$	0.601400				

Table B.43: Specification for a one-part fractional regression model fit to predict the proportion of bot accounts among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Bot Accounts, PFA = 0.30

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-1.201126	0.011812	-101.687455	$0e{+}00$	
Burst	0.122692	0.025741	4.766477	0.000002	0.000005
Age	-0.110871	0.001808	-61.317667	$0e{+}00$	$0e{+}00$
N	3654				
Fit RMSD	0.031452				
$\frac{\text{RMSD}}{\sigma}$	0.600500				

Table B.44: Specification for a one-part fractional regression model fit to predict the proportion of bot accounts among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Bot Accounts, PFA = 0.35

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-1.192450	0.011720	-101.743518	$0e{+}00$	
Burst	0.002606	0.011212	0.232403	0.816225	0.816330
Age	-0.111933	0.001785	-62.710904	$0e{+}00$	$0e{+}00$
N	3654				
Fit RMSD	0.031669				
$\frac{\text{RMSD}}{\sigma}$	0.604641				

Table B.45: Specification for a one-part fractional regression model fit to predict the proportion of bot accounts among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Bot Accounts, PFA = 0.25

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.021641	0.004516	4.791736	0.000002	0.000004	0.413176
Age	-0.014160	0.000260	-54.419922	$0e{+}00$	$0e{+}00$	-0.270340
N	3654					
Fit RMSD	0.031500					
$\frac{\text{RMSD}}{\sigma}$	0.601400					

Table B.46: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the proportion of bot accounts among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Bot Accounts, PFA = 0.30

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.016251	0.003538	4.593828	0.000004	0.000010	0.310274
Age	−0.014131	0.000259	−54.484879	0e+00	0e+00	−0.269789
N	3654					
Fit RMSD	0.031452					
$\frac{\text{RMSD}}{\sigma}$	0.600500					

Table B.47: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the proportion of bot accounts among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Bot Accounts, PFA = 0.35

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.000332	0.001431	0.232268	0.816330	0.816330	0.006344
Age	−0.014267	0.000257	−55.428994	0e+00	0e+00	−0.272399
N	3654					
Fit RMSD	0.031669					
$\frac{\text{RMSD}}{\sigma}$	0.604641					

Table B.48: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the proportion of bot accounts among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Accounts Sharing Low-Credibility Sources, PFA = 0.25

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-3.689571	0.008709	-423.643867	$0e{+}00$	
Burst	0.129077	0.093627	1.378620	0.168012	0.197231
Age	-0.001216	0.001361	-0.893976	0.371335	0.401349
N	3654				
Fit RMSD	0.006111				
$\frac{\text{RMSD}}{\sigma}$	0.997784				

Table B.49: Specification for a one-part fractional regression model fit to predict the proportion of accounts sharing low-credibility sources among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (*N*), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Accounts Sharing Low-Credibility Sources, PFA = 0.30

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-3.697513	0.008771	-421.577952	$0e{+}00$	
Burst	0.174202	0.050155	3.473262	0.000514	0.000926
Age	-0.000402	0.001367	-0.294365	0.768479	0.798085
N	3654				
Fit RMSD	0.006075				
$\frac{\text{RMSD}}{\sigma}$	0.991884				

Table B.50: Specification for a one-part fractional regression model fit to predict the proportion of accounts sharing low-credibility sources among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (*N*), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Accounts Sharing Low-Credibility Sources, PFA = 0.35

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-3.705743	0.008588	-431.509285	$0e{+}00$	
Burst	0.101629	0.014099	7.208030	$6e{-}13$	$2e{-}12$
Age	-0.001470	0.001344	-1.094311	0.273818	0.308625
N	3654				
Fit RMSD	0.006047				
$\frac{\text{RMSD}}{\sigma}$	0.987341				

Table B.51: Specification for a one-part fractional regression model fit to predict the proportion of accounts sharing low-credibility sources among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (*N*), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Accounts Sharing Low-Credibility Sources, PFA = 0.25

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.003247	0.002501	1.298009	0.194284	0.225620	0.530104
Age	-0.000029	0.000032	-0.893444	0.371619	0.401349	-0.004705
N	3654					
Fit RMSD	0.006111					
$\frac{\text{RMSD}}{\sigma}$	0.997784					

Table B.52: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the proportion of accounts sharing low-credibility sources among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Accounts Sharing Low-Credibility Sources, PFA = 0.30

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.004462	0.001392	3.205733	0.001347	0.002385	0.728565
Age	−0.000010	0.000032	−0.294304	0.768526	0.798085	−0.001557
N	3654					
Fit RMSD	0.006075					
$\frac{\text{RMSD}}{\sigma}$	0.991884					

Table B.53: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the proportion of accounts sharing low-credibility sources among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of Accounts Sharing Low-Credibility Sources, PFA = 0.35

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.002481	0.000358	6.922871	4e−12	2e−11	0.405123
Age	−0.000035	0.000032	−1.093137	0.274334	0.308625	−0.005686
N	3654					
Fit RMSD	0.006047					
$\frac{\text{RMSD}}{\sigma}$	0.987341					

Table B.54: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict the proportion of accounts sharing low-credibility sources among accounts created on each day. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Appendix C

Link Inference Methodology

C.1 Data Processing

C.1.1 Timeline Collection for French Terror Tweets

To enhance the Twitter data set of French terror discussions, the Academic Research API [194] was used to collect the timelines of the one thousand most active users of any of the manually coded anti-France hashtags shown in Figure C.1.

#ShameOnYouMacroIf, #The_Dust_Will_Never_Satteled_Down, #مقاطعه_المنتجات_الفرنسيه, #FascistFranceHurtMuslims, #TwitterSuspendMacronAccount, #TwitterSuspendMarcoAccount, #We_Are_The_Ummah_Of_Muhammad_ﷺ, #Pm_Ik_KickoutFranceAmbassador, #WeHateFranceGovernment, #TwitterSuspendMacron, #Boycott_France_Products, #FrancePromotingTerrorism, #StopMacron, #Boycott_Frence_Products, #FranceEnemyOfIslam, #فرنسا_تسيء_لنبي_الأمة, #ماكرون_يسيء_للنبي#, #RTErdoganAnnounceJihadOnFrance, #Boycottmacron, #macronhaddinibil, #Macron_Psychopathe, #JIWomenProtestAgainstFrance, #6مقاطعه_المنتجات_الفرنسيه, #SavaşımızZalimlerle, #ShameOnYouMacro, #The_Dust_Will_Never_Settle_Down, #BoycottFrenchProducts, #Emmanuel_Macron_Real_Terrorist, #TwitterSuspendMarcon, #KickOut_MarcBaréty, #من_سب_نبيافاقتلوه, #MacronApologizeToMuslims, #WeHateFrenchGovernment, #BoycottezLesProduitsFrançais, #4مقاطعه_المنتجات_الفرنسيه, #FransızMallarınaBoykot, #قاطعوا_البضائع_الفرنسية, #boikotprancis, #OICBanFrenchProducts, #boycotfrance, #RacistMacron, #iBycottFranceProducts, #suspendmacronaccount, #bycott_france, #We_hate_French_government, #shameonmacron, #We_Hate_French_President, #FransayaKimDurDiyecek, #We_hate_Frence_government, #BanFranceItems, #MacronGoneMad, #Boycott_French, #france_will_down, #baycott_france, #KickOutFrenchAmbassador, #WeBoycott_FranceProduct, #3فرانسی_مصنوعات_کا_مکمل_بائیکاٹ_کیجئے#, مقاطعه_المنتجات_الفرنسيه, #We_Hate_France_President, #Boycottez_les_produits_français, #फ्रांस_माफी_मांग, #shameonfrance, #FranceBoycott, #ShameOnMacronExtremist, #مقاطعه_متاجر__كارفور_الفرنسيه, #Boycott_France, #BanFranceproducts, #boycotfrenchproducts, #ماكرون_خنزير_فرنسا, #MacronNeedsMentalCheck, #FranceMustApologize, #خان_صاحب_سفير_فرانس_نكاليں, #FranceShameOnYou, #BoycotFranceProducts, #AhlaksızFransa, #suspendmacroaccount, #boycottfrenchgoods, #boycottfrench, #LoveJihaad, #TwitterSuspendMacroAccount, #مقاطعه_المنتجات_الفرنسية, #Boycottez_les_produits_francais, #BOYCOTTFRANCEPRODUCT, #WeBoycott_FranceProducts, #إلا_رسول_الله_يافرنسا, #boycot_france, #BoikotProdukPerancis, #boycottfranchproducts, #aquaprodukprancis, #2مقاطعه_المنتجات_الفرنسيه, #Macron_Needs_To_See_a_Psychiatrist, #MacronSaySorryToMuslims, #فرانس_کا_سفیر_نکالو, #TakeActionAgainstFrance, #BoycottFrenchProductsforever, #boycottfrance, #مقاطعة_منتجات_فرنسا, #ফ্রান্স_পন্য_বর্জন_করুন, #فرانس_سے_تعلق_ختم_کرو, #bycotfranceproducts, #boykottfrance, #BoikotprodukPrancis, #Boycott_France_Government, #نیازی_فرانس_کاسفیرنکالو, #boycottfranceproducts, #macronthedevil, #آقا_دو_جہاں_پر_جان_قربان, #FransaBoykot, #PKSKecamMacron, #France_SpreadingHate, #BoycottFrenceProducts, #get_out_frence_ambassador, #فرانس_کے_خلاف_جہادکااعلان_کرو, #fransamallarınaboykot, #BycottFrenchProducts, #مقاطعة_المنتجات_الفرنسية, #We_Hate_France_Government, #Boycott_french_product, #EmbassadorWillLeavePak, #RemoveFranceFromUNSC, #terroristfrance, #Boycott_French_Products, #Boykotfransa, #HaddiniBilMacron, #مقاطعة_البضائع_الفرنسية, #ShameOnYouMacron, #قاطعوا_المنتجات_الفرنسية, #FranceEnemyOfİslam, #BoycottFrenchProduct, #BycottFrance, #bycottfranceproducts, #ShameOnYouFrance

Figure C.1: The anti-France hashtags used to collect the French terror timelines.

C.1.2 Topic Construction for Islamic Network Forum Data

The topic of each original post in the Islamic Network forum [197] was categorized if it contained any of the following keywords.

Guantanmo Bay

guantanamo, guantanamos, gitmo, gitmos, gtmo, leeward, mccalla, detainee, detainees, detaine, detaines

Afghanistan

afghanistan, afghanistan's, afghanistans, afganistan, afganistans, afghani, afghanis, afgani, afganis, afghan, afgan, pashtun, pashtuns, gwot, mazar i sharif, marmal, tora bora, kabul, hamid karzai, karzai, kandahar, bagram, dwyer, eggers, leatherneck, qargha, forward operating base, operating base, fenty, gardez, geronimo, ghanzi, maiwand

Israel-Palestine

israel, israels, israeli, israelis, palestine, palestines, palestinian, palestinians, jerusalem, jerusalems, west bank, west banks, gaza, gazas, haifa, haifas, tel aviv, tel avivs, ramallah, ramallahs, ramalah, ramalahs, hebron, hebrons, jericho, jerichos, fatah, fatahs, hamas, hamass, jenin, jenins

Chechnya

chechnya, chechnyas, chechen, chechnyan, chechnyans, chechnia, chechnias, chechnian, chechnians, grozny, groznys, caucusus, caucususs, caucases, caucasess, ciscaucasia, ciscaucasias, ciscaucasian, ciscaucasians, ingush, ingushs, ingushetia, ingushetias, ingushetian, ingushetians, chechneia

Kashmir

kashmir, kashmirs, kashmiri, kashmiris, jammu, ladakh

C.2 Link Inference

C.2.1 Tensor Perturbation Method

Algorithm C.1 defines the tensor perturbation method (TPM).

C.2.2 Parameter Optimization

Table C.1 provides more details on the parameters used for the link prediction algorithms and the values tested in the cross-validation process.

Algorithm C.1: Tensor perturbation method (TPM). Let $\boldsymbol{\mathcal{T}}$ be the input network, p be the proportion of edges to add to the network during each iteration, t be the number of iterations, $\boldsymbol{k}$ be the set of dimensions to *not* compress, ϵ be the allowable error for the Colibri-S algorithm, and c be the number of column samples for the Colibri-S algorithm.

Let *rand* : $(x \in \mathbb{N} | x \geq 1) \rightarrow (y \in \mathbb{N} | y \in [1, x])$ be a function that selects a random natural number in the interval $[1, x]$.
Let *Colibri-S* refer to the Colibri-S algorithm [190].
Let *Tucker* refer to the incomplete non-negative Tucker decomposition [112].
Let $\boldsymbol{\mathcal{T}}_i$ indicate unfolding [112] the tensor $\boldsymbol{\mathcal{T}}$ along dimension i.

Input: $\boldsymbol{\mathcal{T}} \in \mathbf{N}_{n_1 \times n_2 \times \cdots \times n_d}$, $p > 0$, $t \in \mathbb{N}$, $\boldsymbol{k} = \{x | x \in \mathbb{N} \wedge x \in [1, d]\}$, $\epsilon > 0$, $c \in \mathbb{N}$
Output: $\boldsymbol{\mathcal{S}} \in \mathbb{R}_{n_1 \times n_2 \times \cdots \times n_d}$

$\boldsymbol{\mathcal{S}} \leftarrow \mathbf{0}_{n_1 \times n_2 \times \cdots \times n_d}$
$e \leftarrow \lceil p \sum_{i_d=1}^{n_d} \cdots \sum_{i_2=1}^{n_2} \sum_{i_1=1}^{n_1} \boldsymbol{\mathcal{T}}_{i_1, i_2, \ldots, i_d} \rceil$
$\boldsymbol{r} \leftarrow (n_1, n_2, \ldots, n_d)$
for $i \leftarrow 1$ **to** d **do**
  if $i \notin \boldsymbol{k}$ **then**
    $\boldsymbol{L}, \boldsymbol{M}_{\tilde{c} \times \tilde{c}}, \boldsymbol{R} \leftarrow$ *Colibri-S* $(\boldsymbol{\mathcal{T}}_i, \epsilon, c)$
    $r_i \leftarrow \tilde{c}$
  end
end
for $i \leftarrow 1$ **to** t **do**
  $\widetilde{\boldsymbol{\mathcal{T}}} \leftarrow \boldsymbol{\mathcal{T}}$
  for $j \leftarrow 1$ **to** e **do**
    $m_1, m_2, \ldots, m_d \leftarrow$ *rand* (n_1), *rand* (n_2), $\ldots$, *rand* (n_d)
    while $m_{d-1} = m_d$ **do**
      $m_{d-1}, m_d \leftarrow$ *rand* (n_{d-1}), *rand* (n_d)
    end
    $\widetilde{\mathcal{T}}_{m_1, m_2, \ldots, m_d} \leftarrow \widetilde{\mathcal{T}}_{m_1, m_2, \ldots, m_d} + 1$
  end
  $\boldsymbol{\mathcal{C}}, \boldsymbol{U}_1, \boldsymbol{U}_2, \ldots, \boldsymbol{U}_d \leftarrow$ *Tucker* $\left(\widetilde{\boldsymbol{\mathcal{T}}}, \boldsymbol{r}\right)$
  $\boldsymbol{\mathcal{S}} \leftarrow \boldsymbol{\mathcal{S}} + (\boldsymbol{\mathcal{C}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \times_3 \cdots \times_d \boldsymbol{U}_d)$
end
$\boldsymbol{\mathcal{S}} \leftarrow \boldsymbol{\mathcal{S}} / t$

Algorithm	Dimension	Directed	Constants	Optimized
Common neighbors (CN)	Simple	No	Input symmetrized	
Adamic-Adar (AA)	Simple	No	Input symmetrized	
Resource allocation (RA)	Simple	No	Input symmetrized	
Structural perturbation method (SPM)	Simple	Yes [211]	Edge weights removed: 10%	
Matrix completion (MC)	Simple	Yes		λ: 0.02, 0.05, 0.08, 0.11, 0.14, 0.17, 0.2, 0.23, 0.26, 0.29, 0.32, or 0.35
Matrix perturbation and decomposition (MPD)	Simple	Yes	Edge weights removed: 10%; number of iterations: 10	λ: 0.07, 0.1, 0.13, 0.16, 0.19, 0.22, 0.25, 0.28, 0.31, 0.34, or 0.37
Multilayer random walk (MLRW)	Multiplex	Yes		Layer normalization in real multiplex trials: none or maximum
Fourth-order non-negative canonical polyadic decomposition (NCP4)	Dynamic	Yes		Rank: 10, 25, 50, 75, or 100; layer normalization in real multiplex trials: none or maximum
Non-negative matrix factorization perturbation (NMFP)	Simple	Yes	Edge weights added: 10%; number of iterations: 10	
Tensor perturbation method (TPM)	Dynamic multiplex	Yes	Layer and time dimensions kept constant during compression; edge weights added: 10%; number of iterations: 10	Layer normalization in real multiplex trials: none or maximum

Table C.1: The values tested in the five-fold cross-validation process used to optimize the parameters for the link prediction algorithms. For all uses of Colibri-S, the tolerance was set to $\epsilon = 10^{-12}$ and the number of column resamplings was set to $10n$, where n is the number of nodes in the network.

Appendix D

Simulated Missing Link Inference Results

D.1 Multiplex Stylized Networks

D.1.1 Small-World Networks

Figures D.1 through D.4 show the results for inferring missing links in multiplex small-world networks.

D.1.2 Scale-Free Networks

Figures D.5 through D.8 show the results for inferring missing links in multiplex scale-free networks.

D.1.3 Uniformly Random Networks

Figures D.9 through D.12 show the results for inferring missing links in multiplex uniformly random networks.

D.2 Dynamic Multiplex Stylized Networks

D.2.1 Small-World Networks

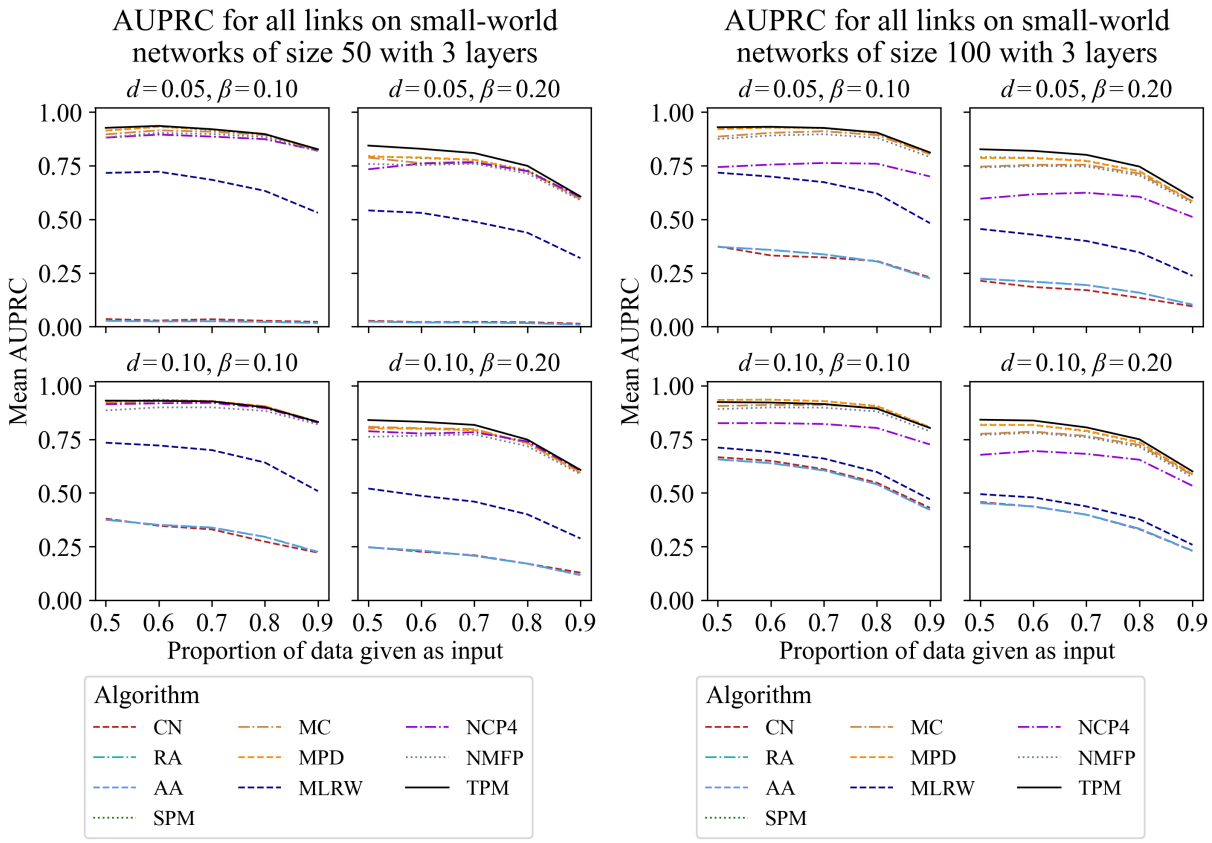
Figures D.13 through D.16 show the results for inferring missing links in dynamic multiplex small-world networks.

D.2.2 Scale-Free Networks

Figures D.17 through D.20 show the results for inferring missing links in dynamic multiplex scale-free networks.

D.2.3 Uniformly Random Networks

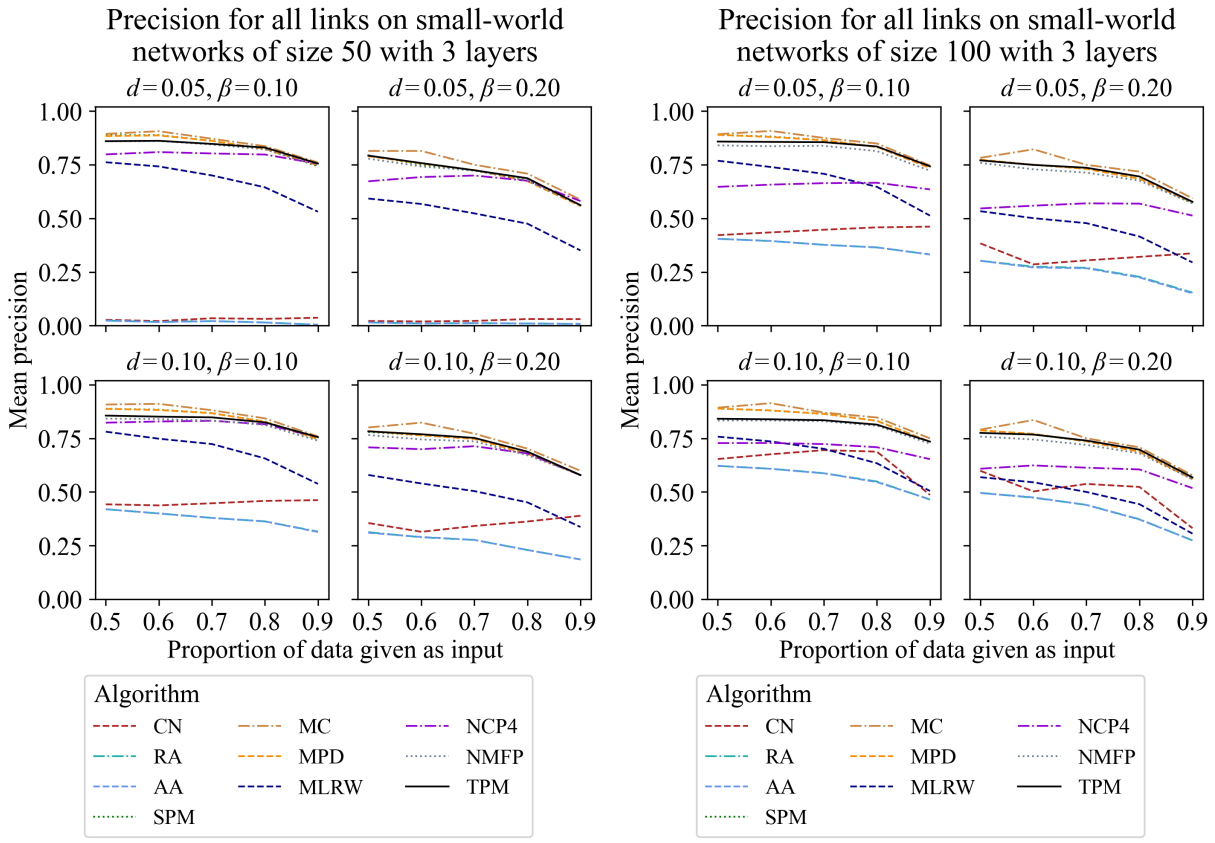
Figures D.21 through D.24 show the results for inferring missing links in dynamic multiplex uniformly random networks.



(a) Fifty-node networks

(b) Hundred-node networks

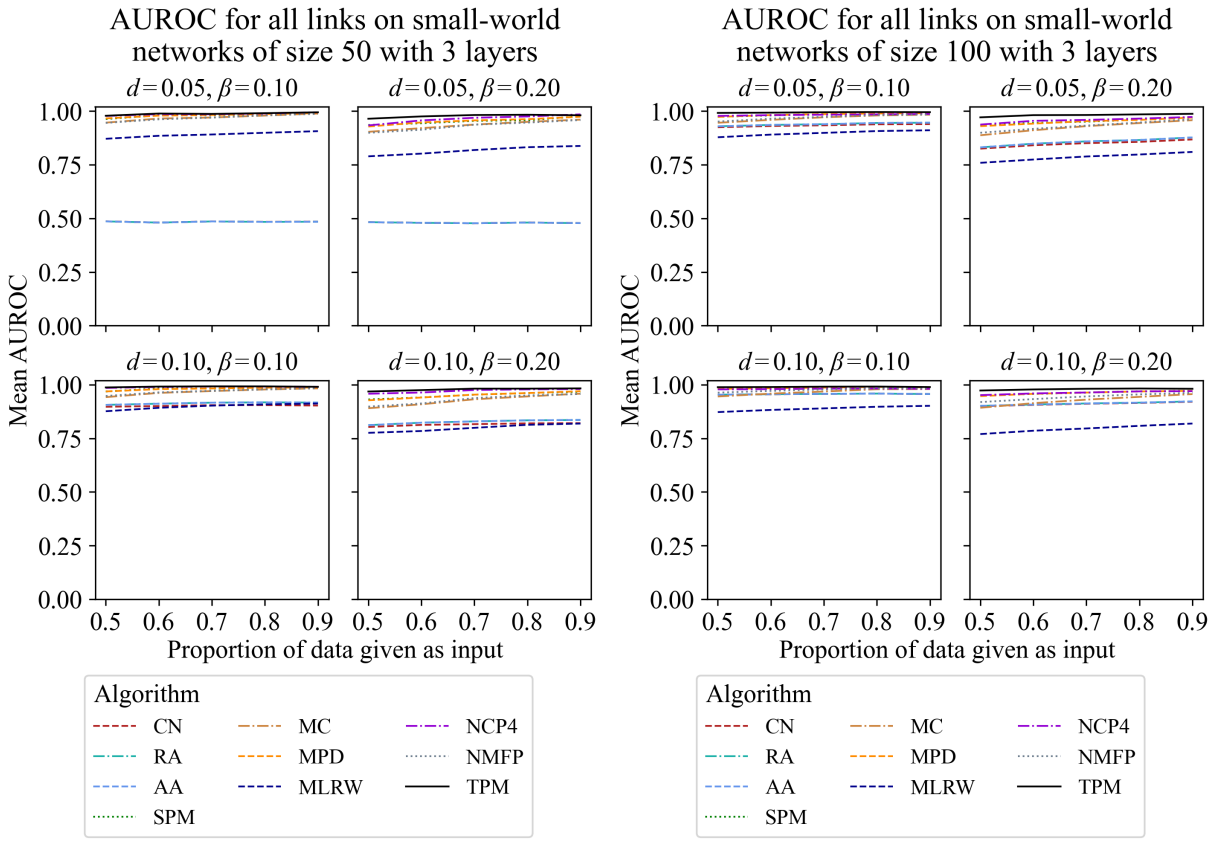
Figure D.1: Area under the precision-recall curve (AUPRC) for inferring any missing links in multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

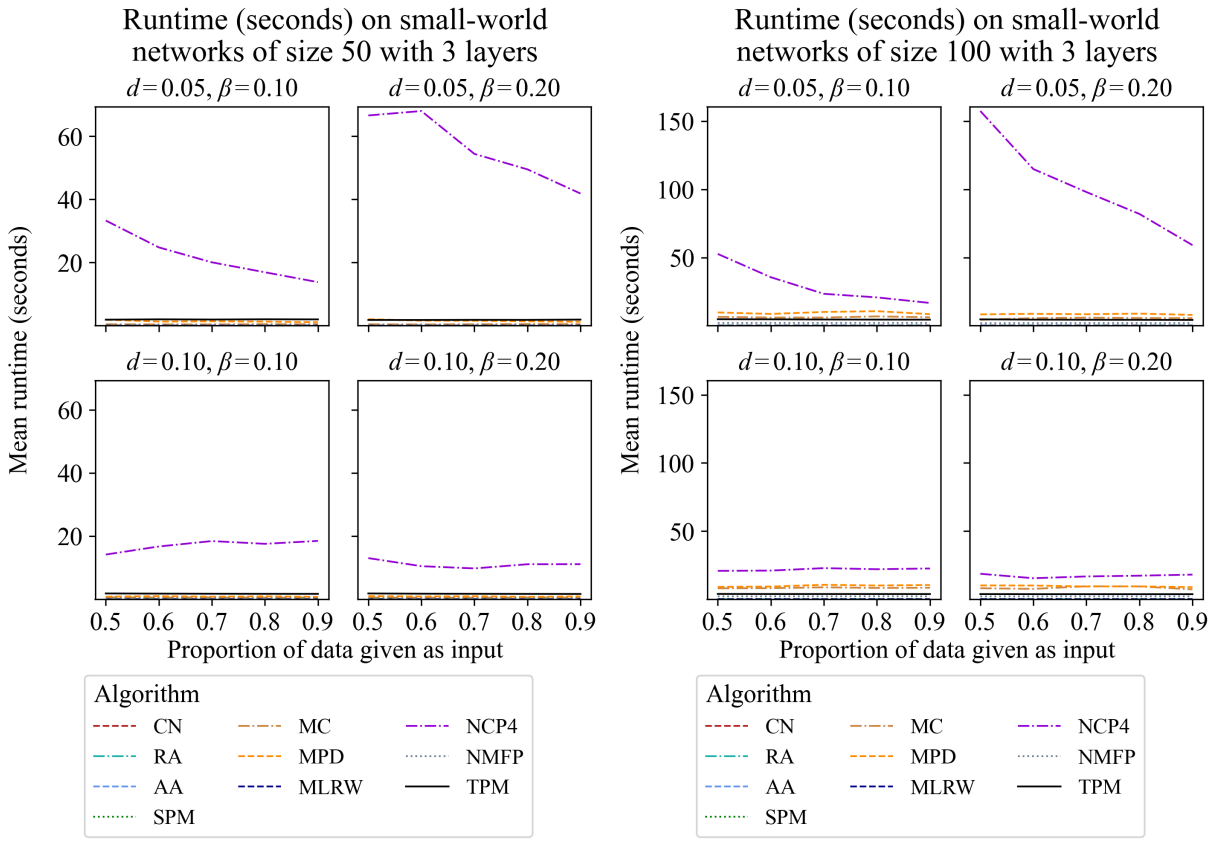
Figure D.2: Precision for inferring any missing links in multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

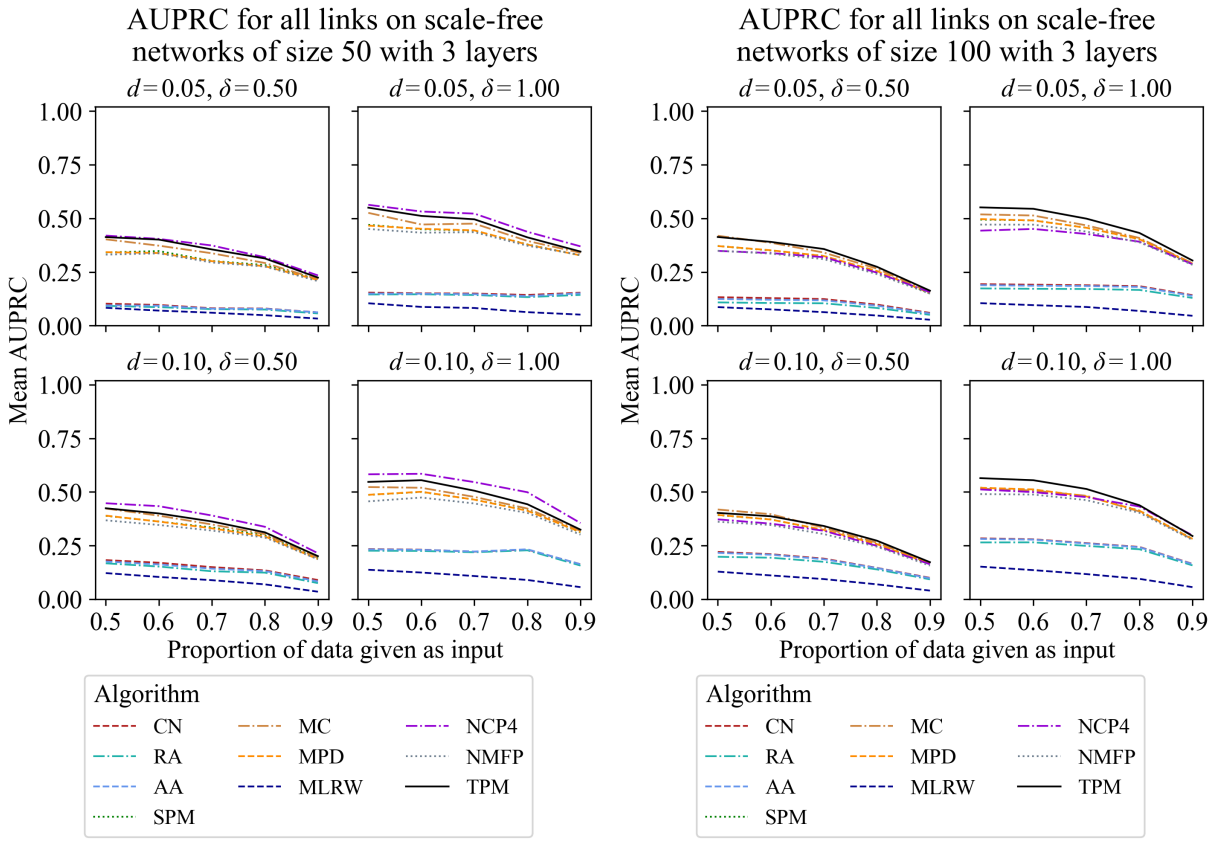
Figure D.3: Area under the receiver operating characteristic (AUROC) for inferring any missing links in multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

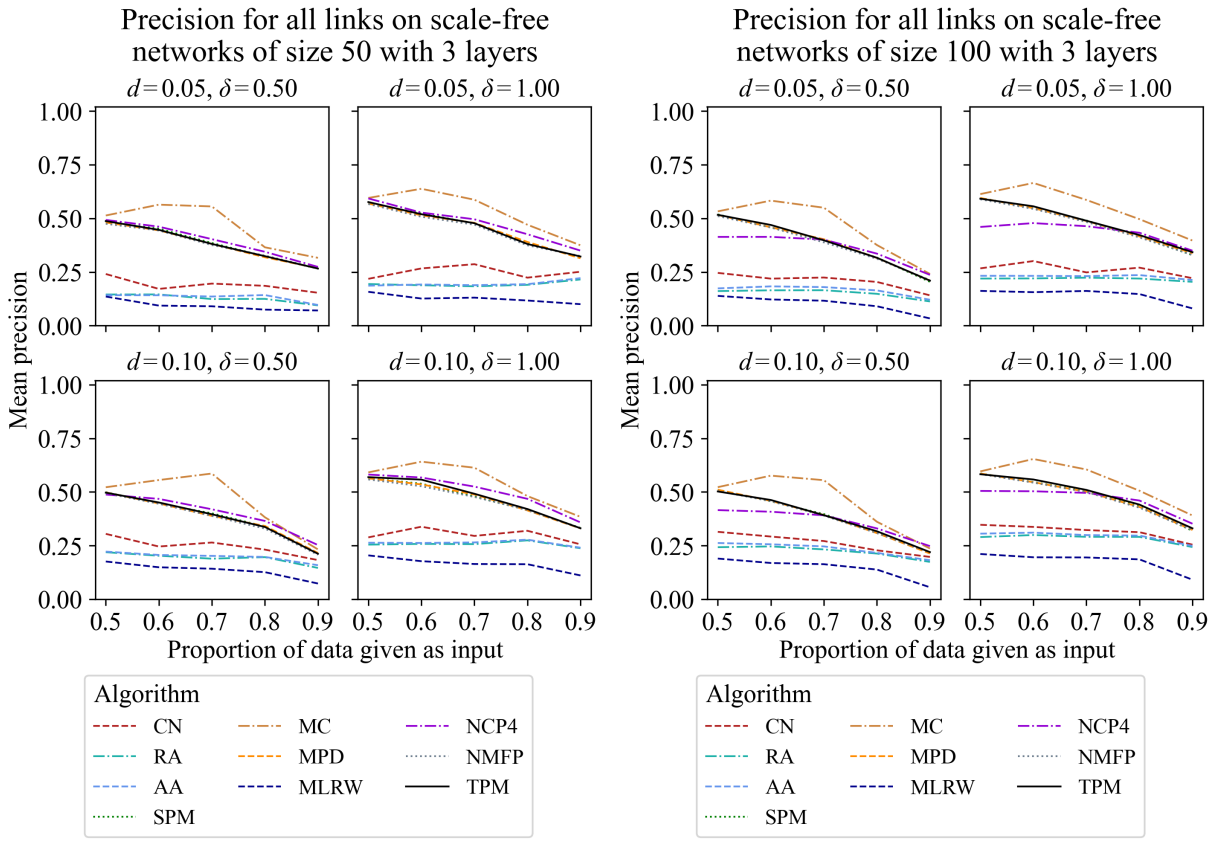
Figure D.4: Runtime for inferring any missing links in multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

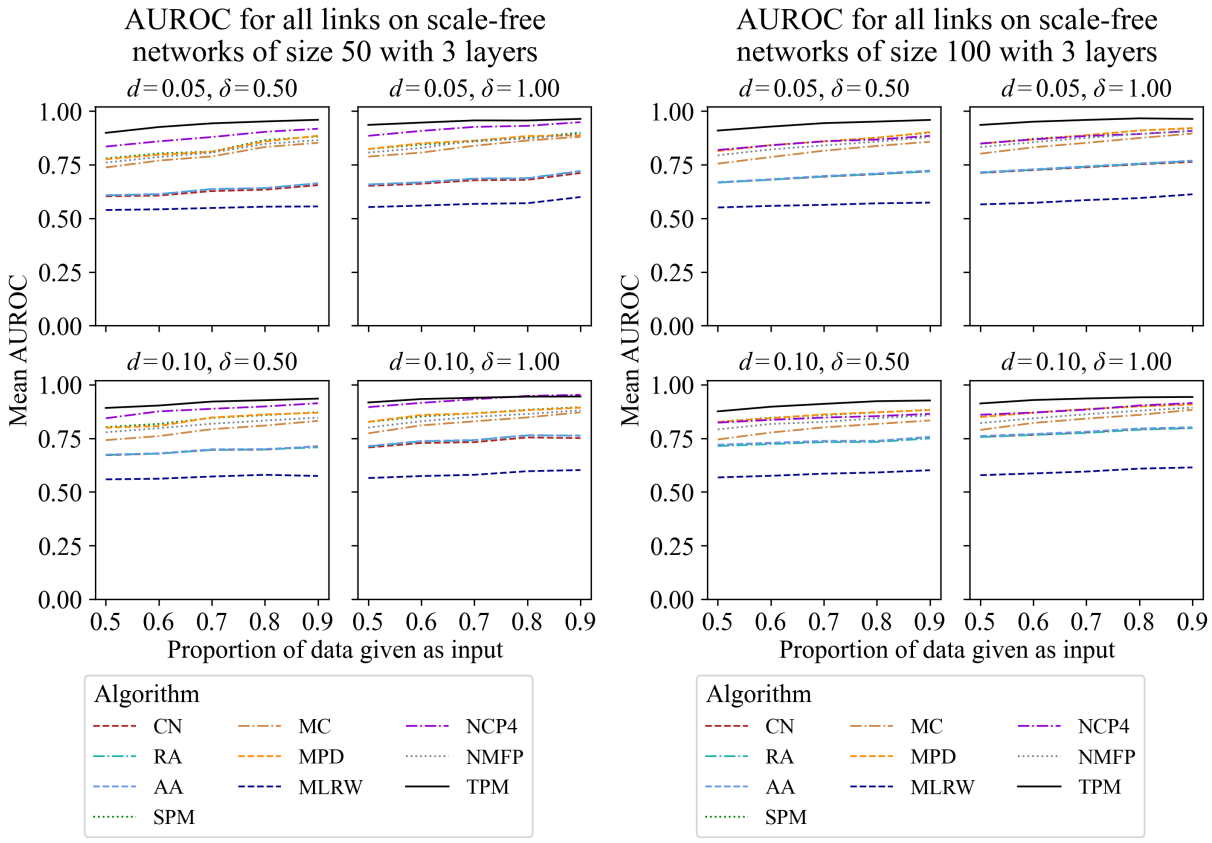
Figure D.5: Area under the precision-recall curve (AUPRC) for inferring any missing links in multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

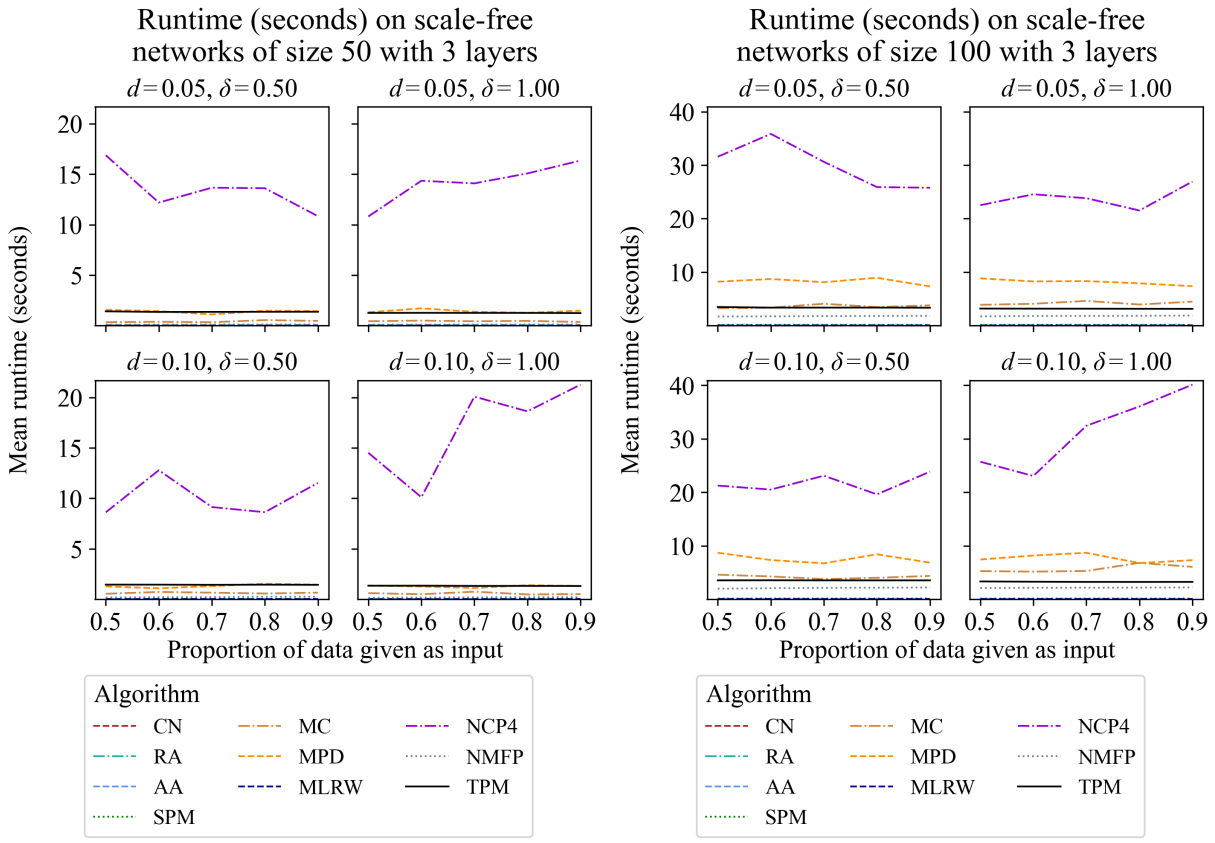
Figure D.6: Precision for inferring any missing links in multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

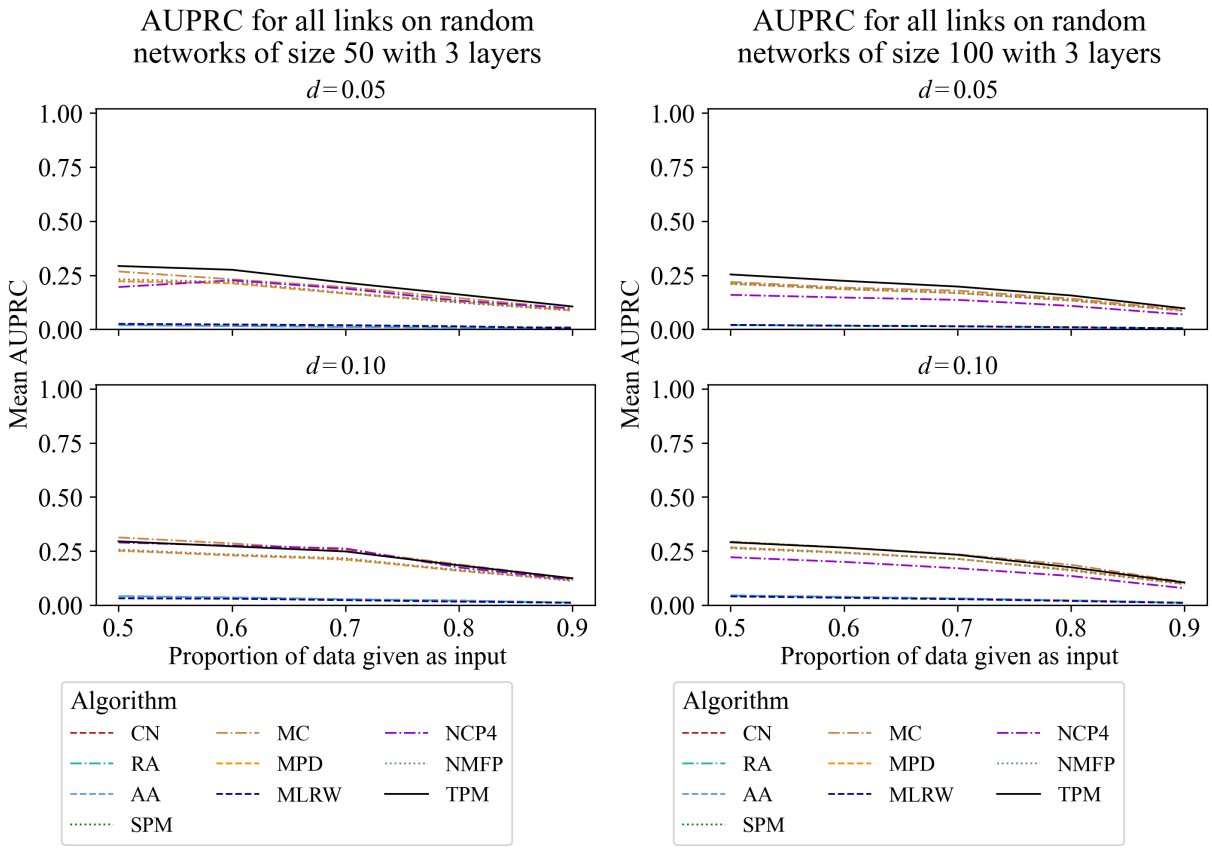
Figure D.7: Area under the receiver operating characteristic (AUROC) for inferring any missing links in multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

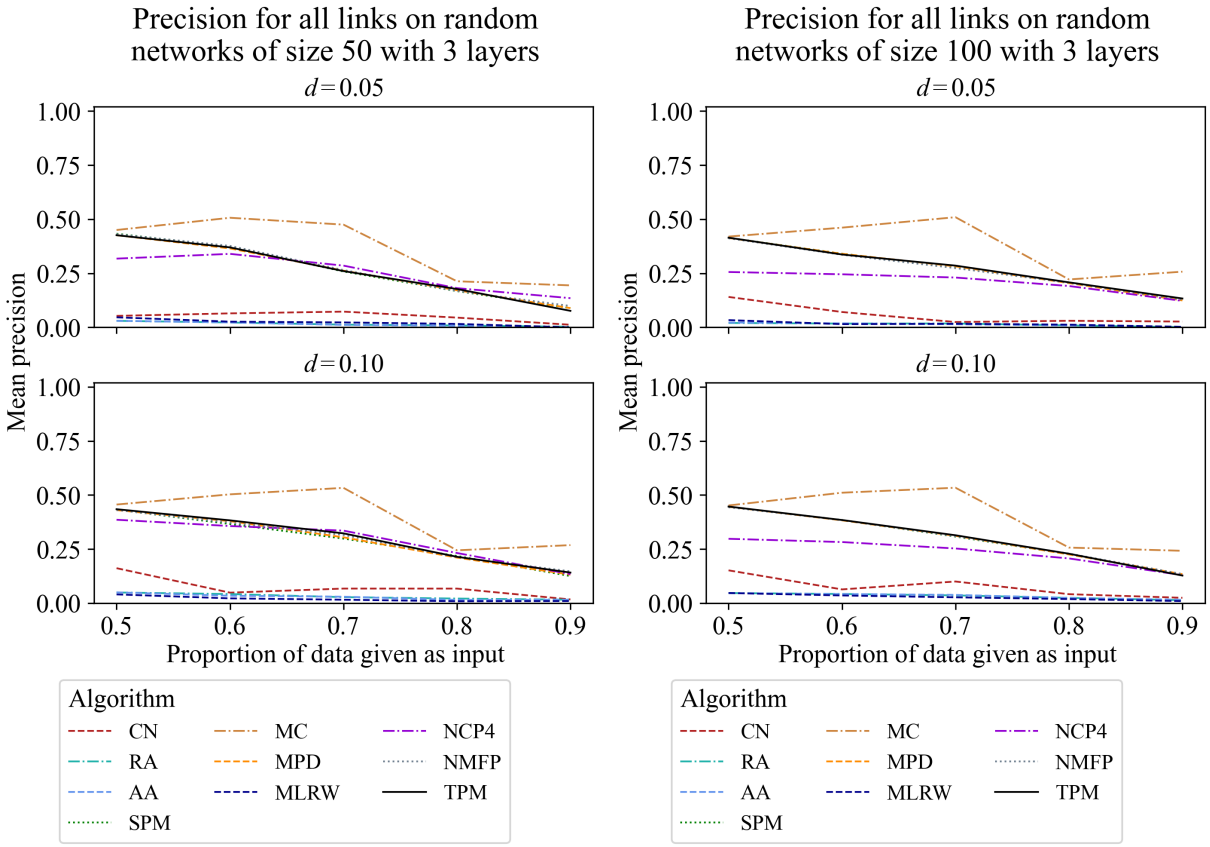
Figure D.8: Runtime for inferring any missing links in multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

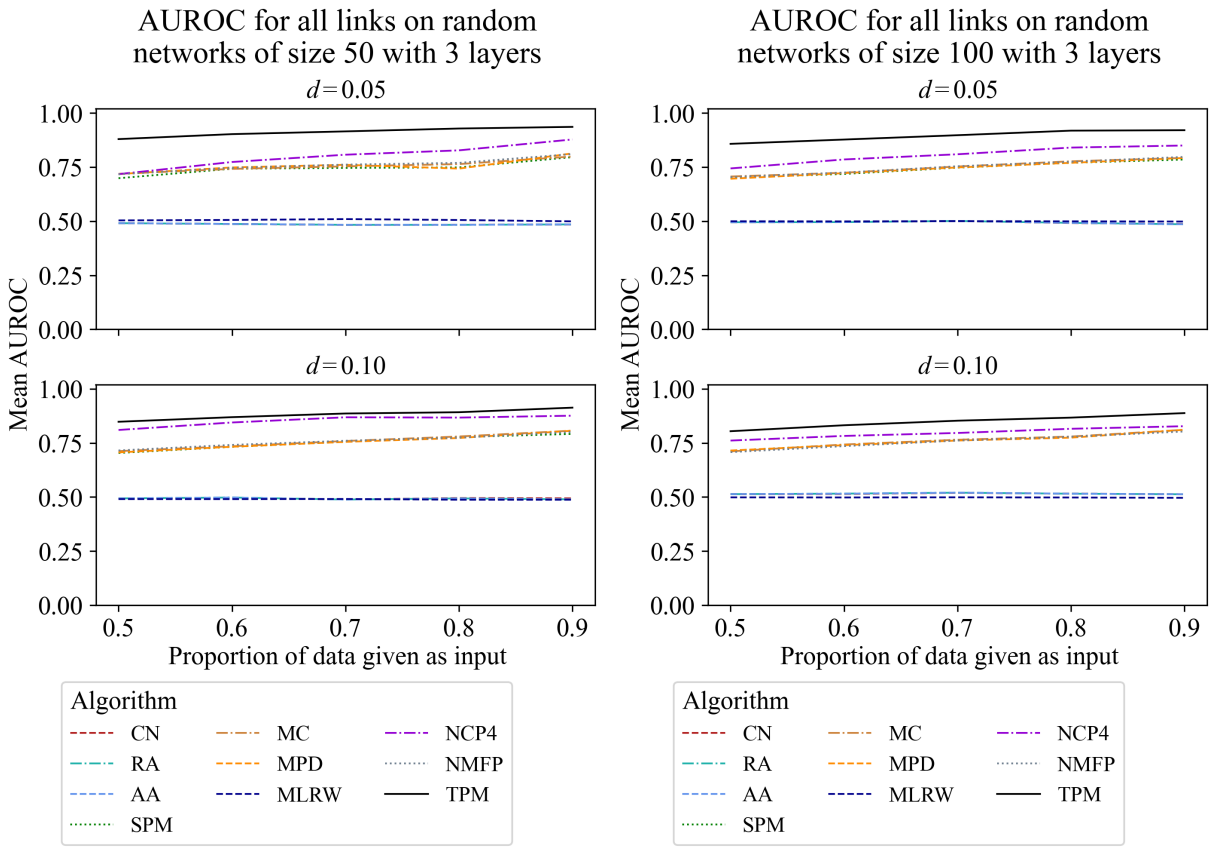
Figure D.9: Area under the precision-recall curve (AUPRC) for inferring any missing links in multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

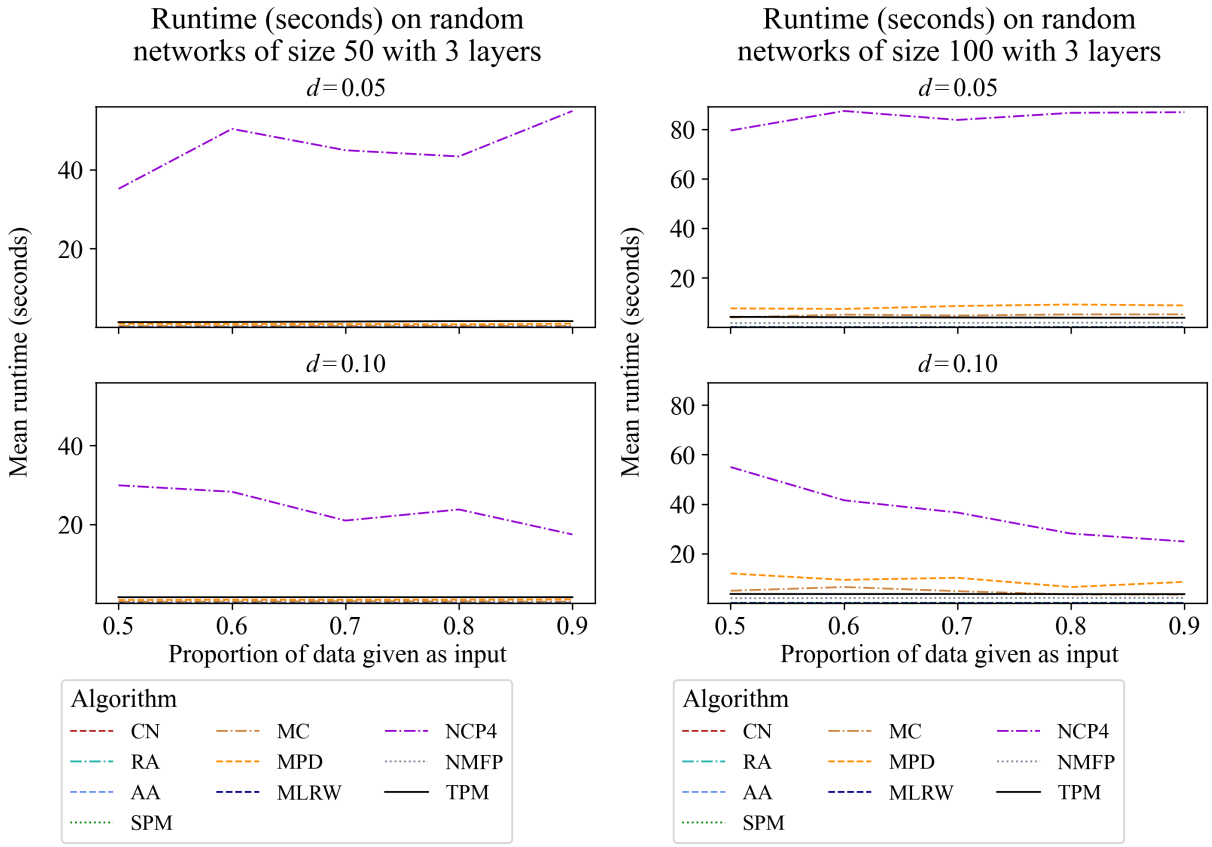
Figure D.10: Precision for inferring any missing links in multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

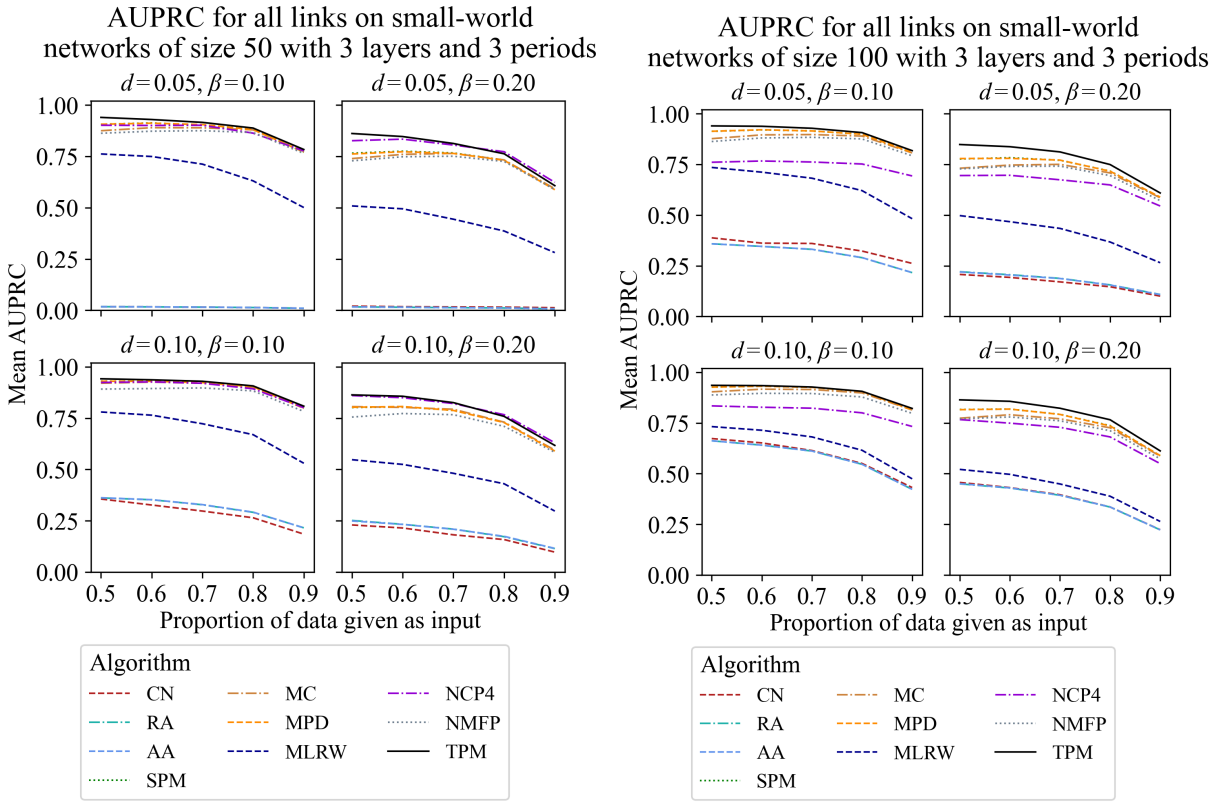
Figure D.11: Area under the receiver operating characteristic (AUROC) for inferring any missing links in multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

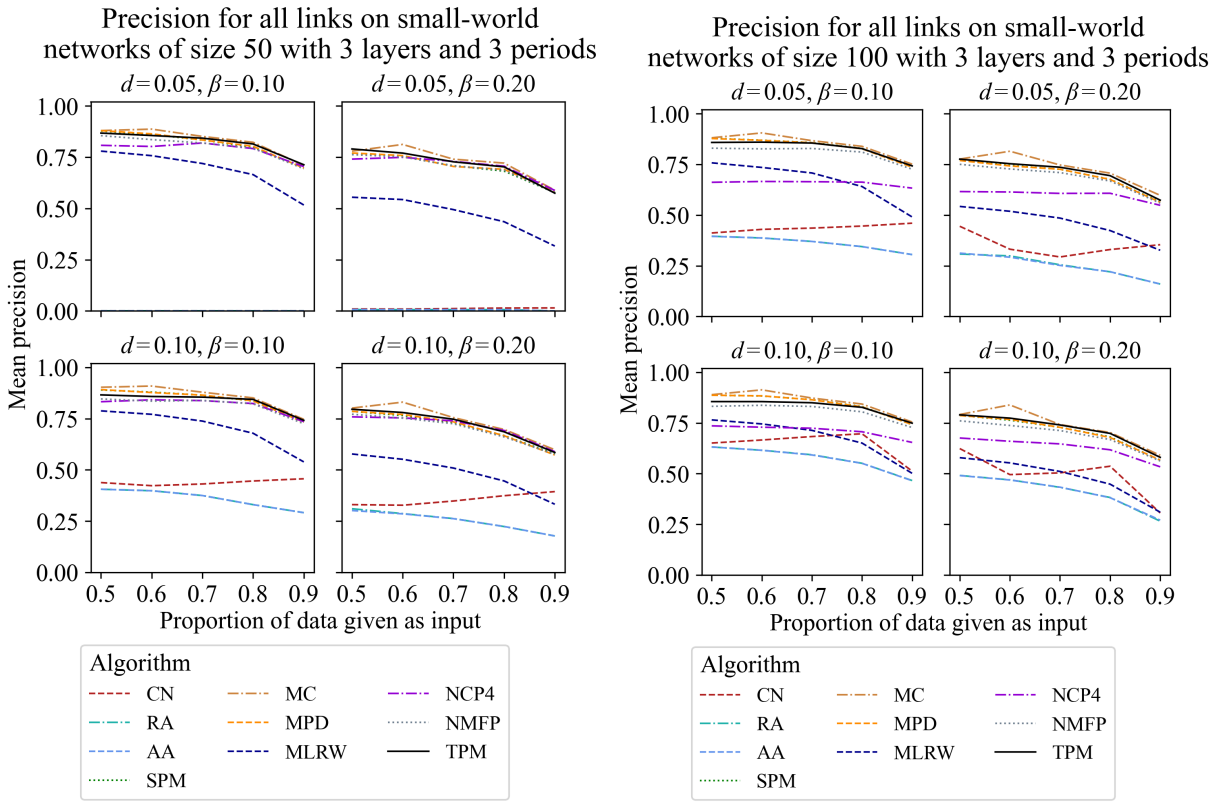
Figure D.12: Runtime for inferring any missing links in multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

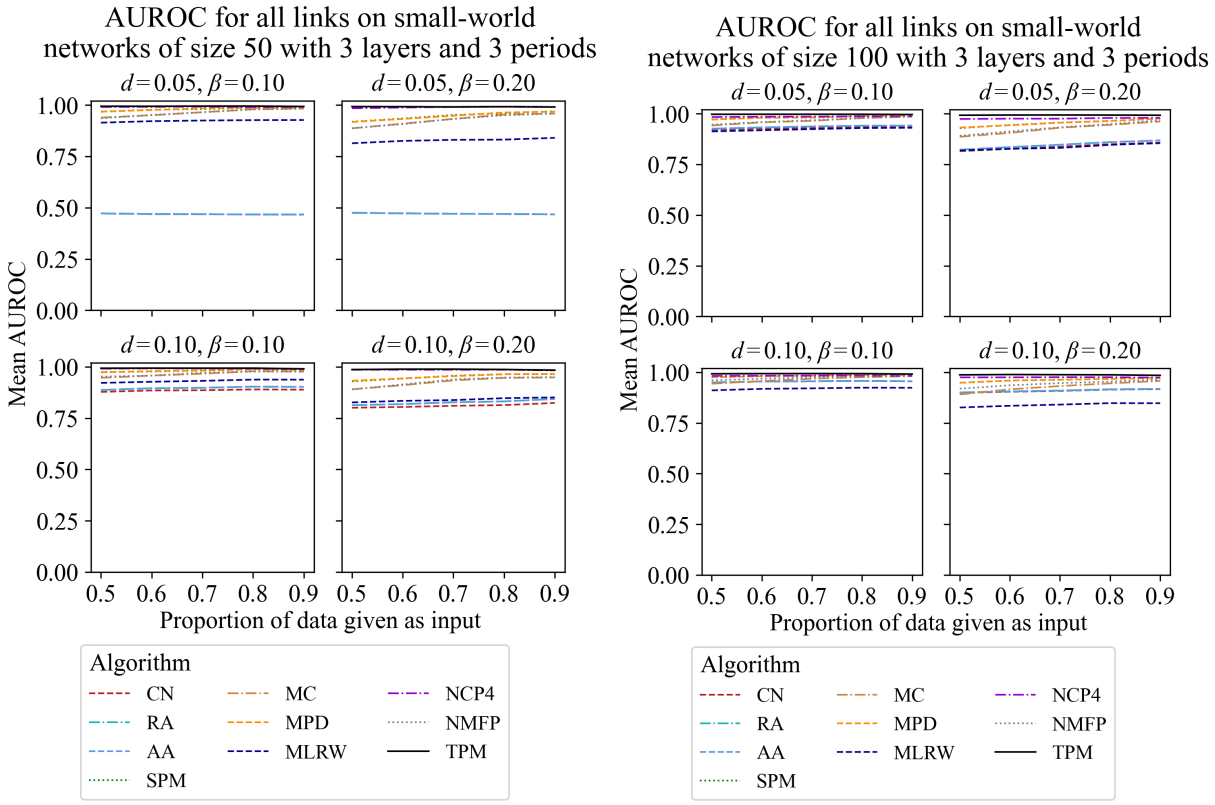
Figure D.13: Area under the precision-recall curve (AUPRC) for inferring any missing links in dynamic multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

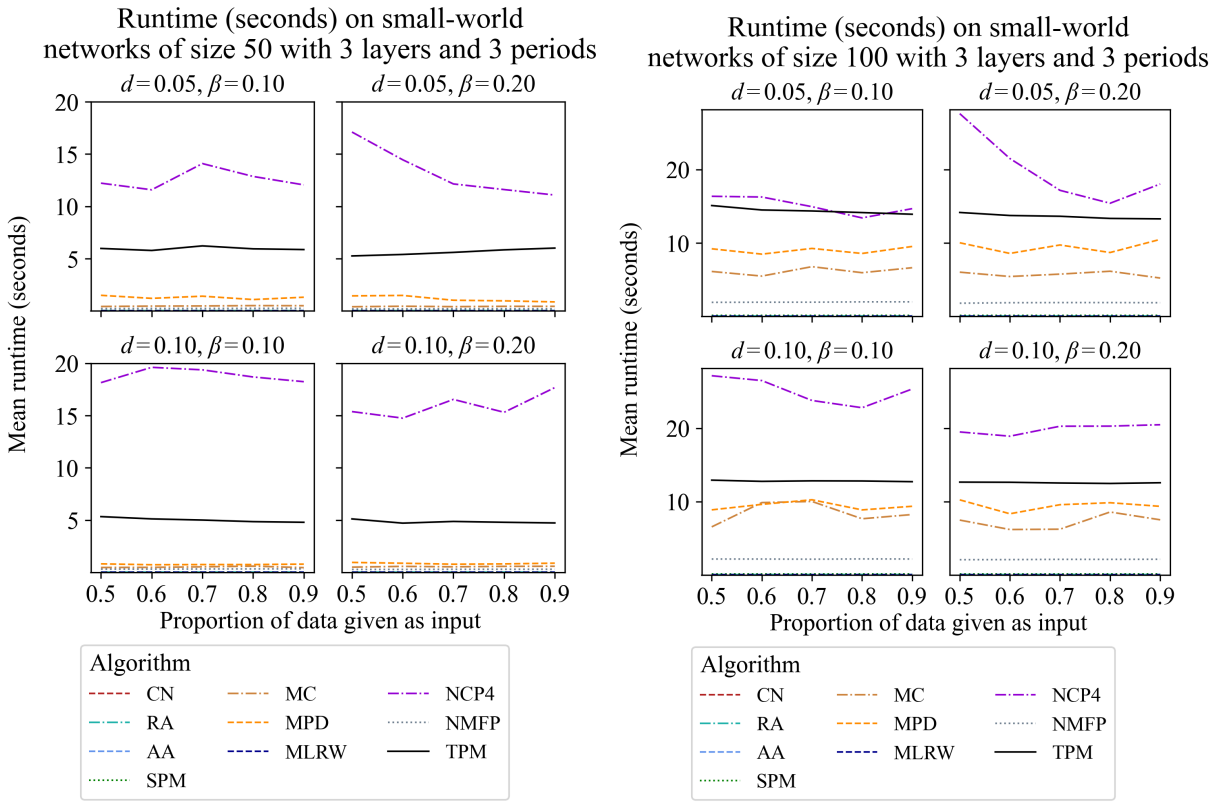
Figure D.14: Precision for inferring any missing links in dynamic multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

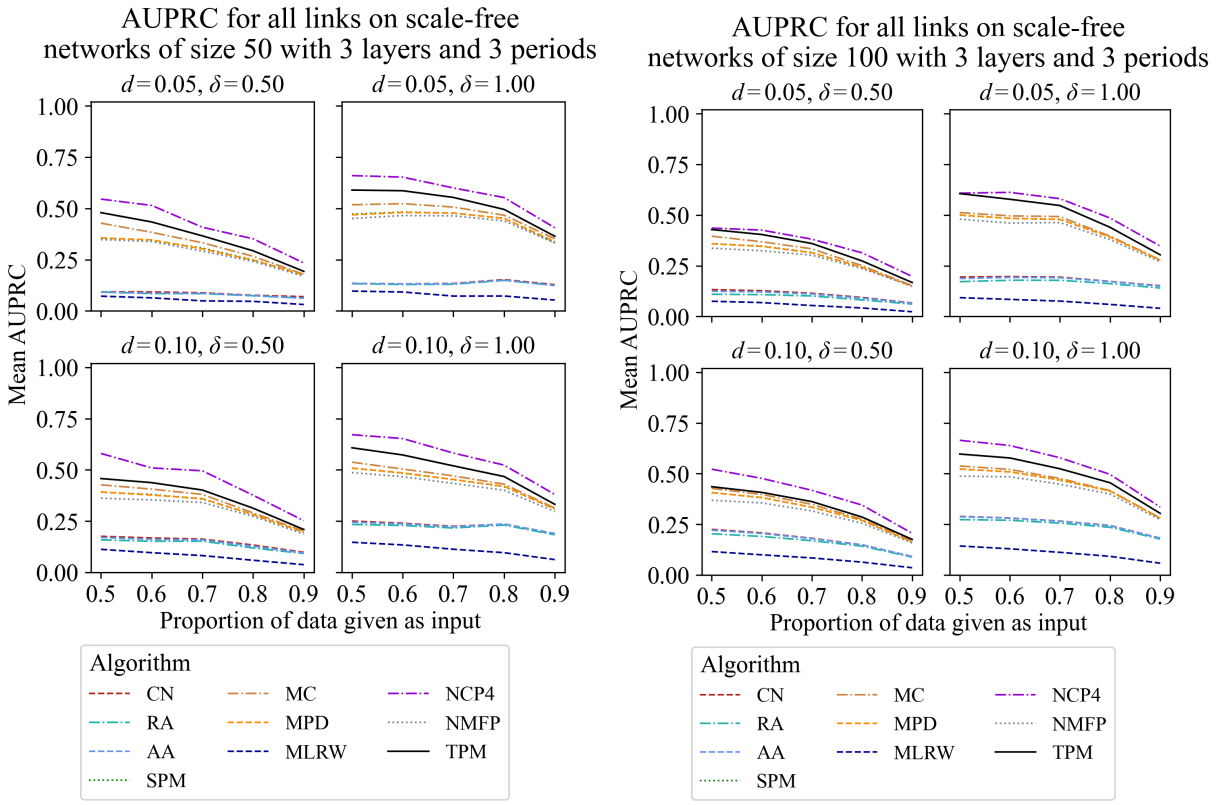
Figure D.15: Area under the receiver operating characteristic (AUROC) for inferring any missing links in dynamic multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

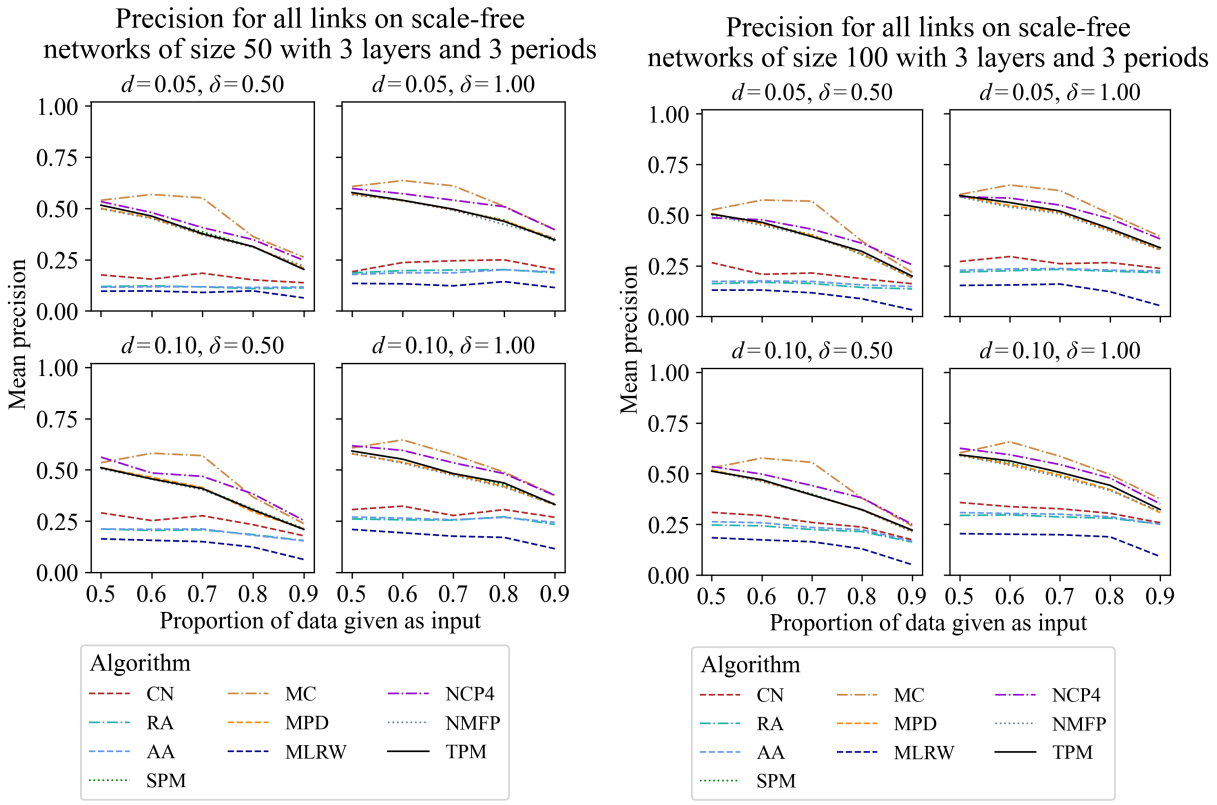
Figure D.16: Runtime for inferring any missing links in dynamic multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

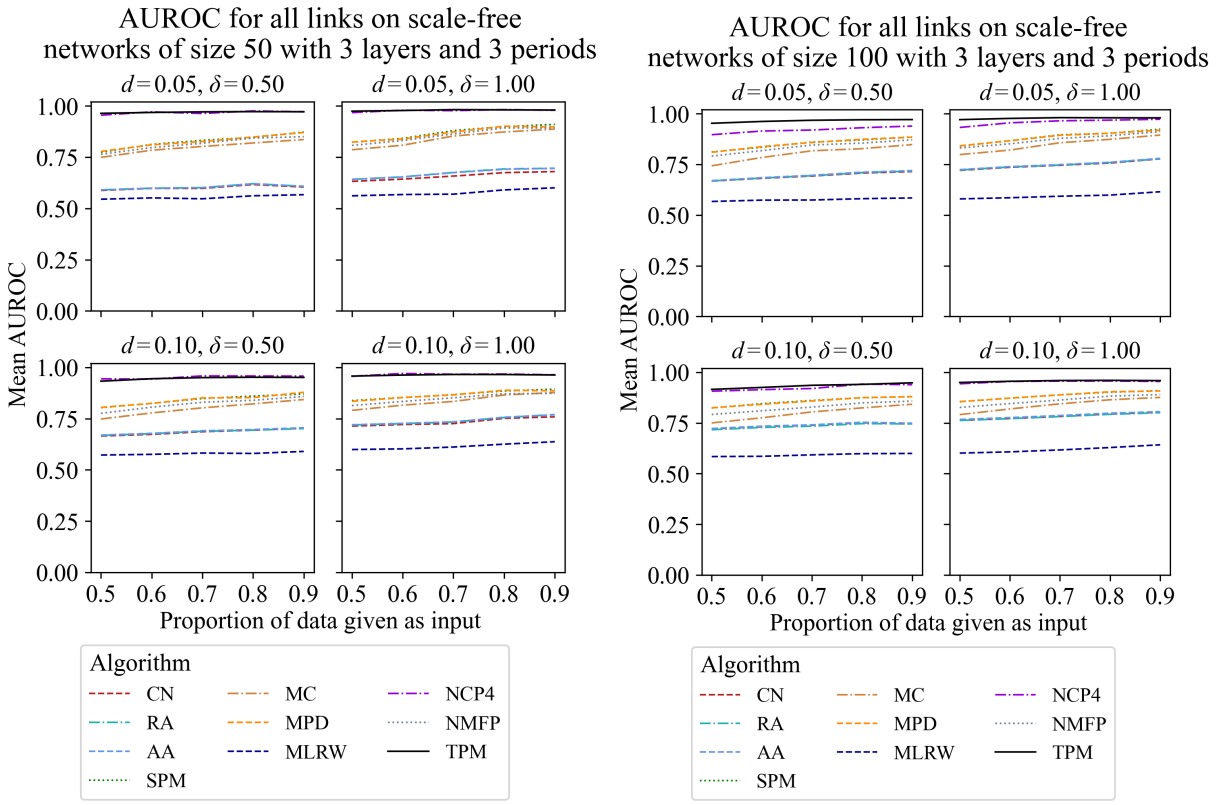
Figure D.17: Area under the precision-recall curve (AUPRC) for inferring any missing links in dynamic multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

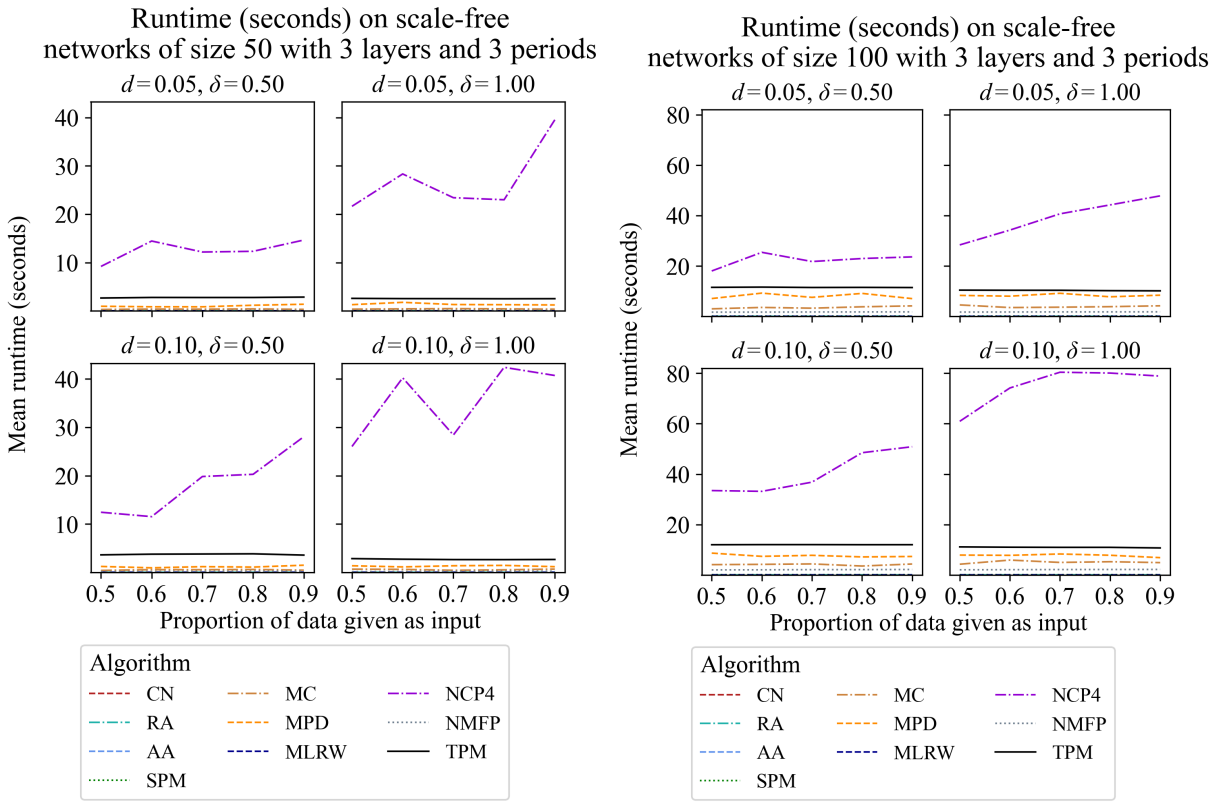
Figure D.18: Precision for inferring any missing links in dynamic multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

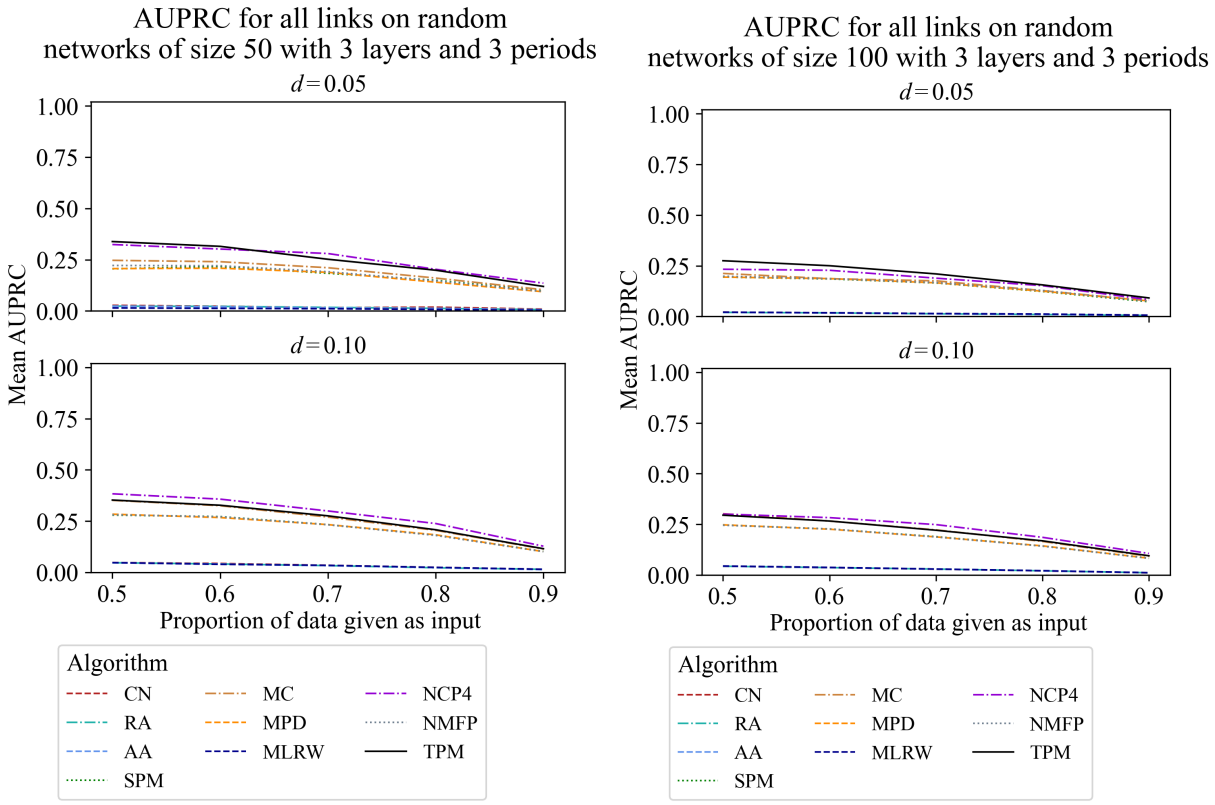
Figure D.19: Area under the receiver operating characteristic (AUROC) for inferring any missing links in dynamic multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

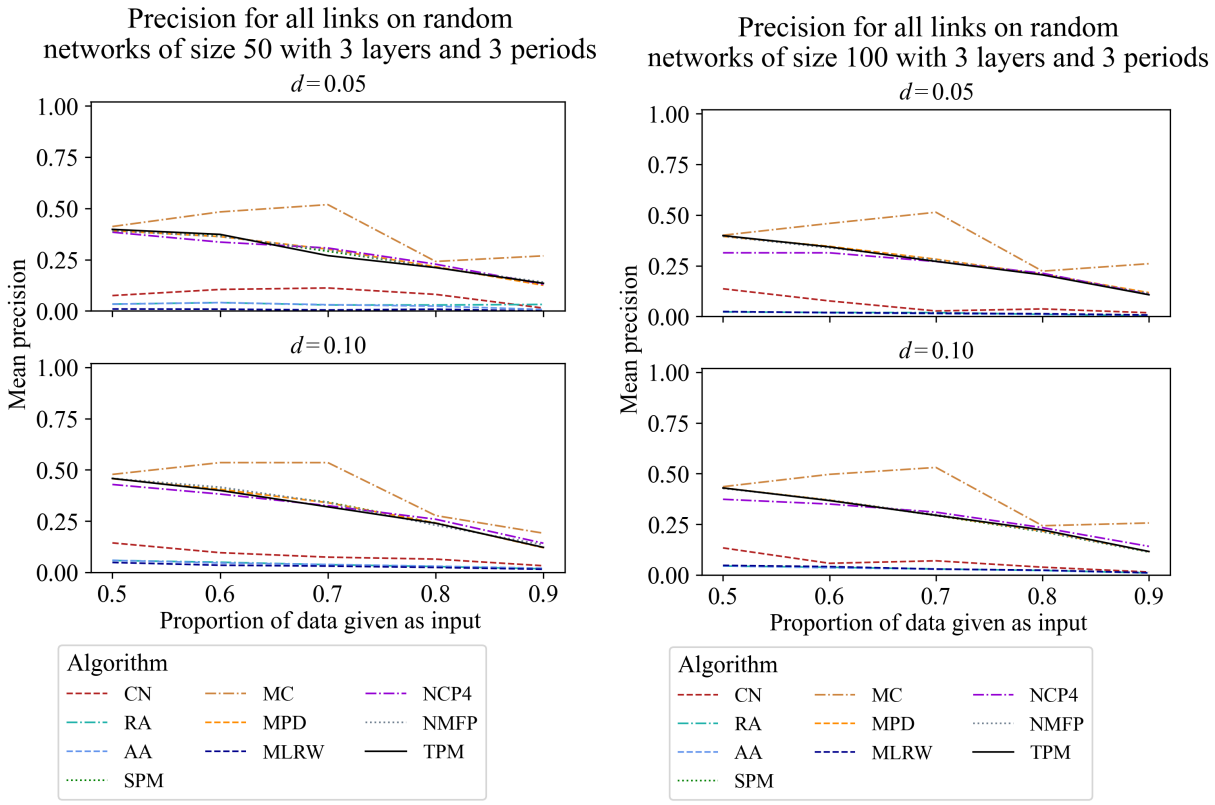
Figure D.20: Runtime for inferring any missing links in dynamic multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

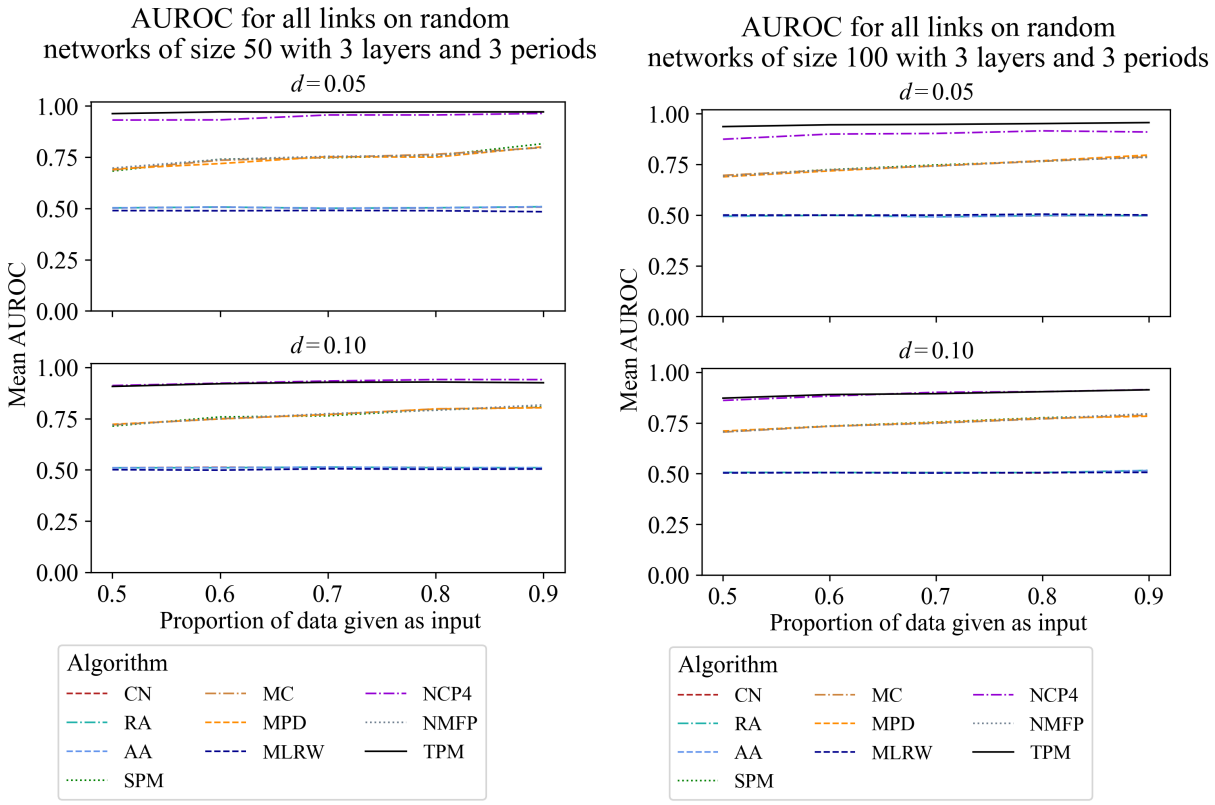
Figure D.21: Area under the precision-recall curve (AUPRC) for inferring any missing links in dynamic multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

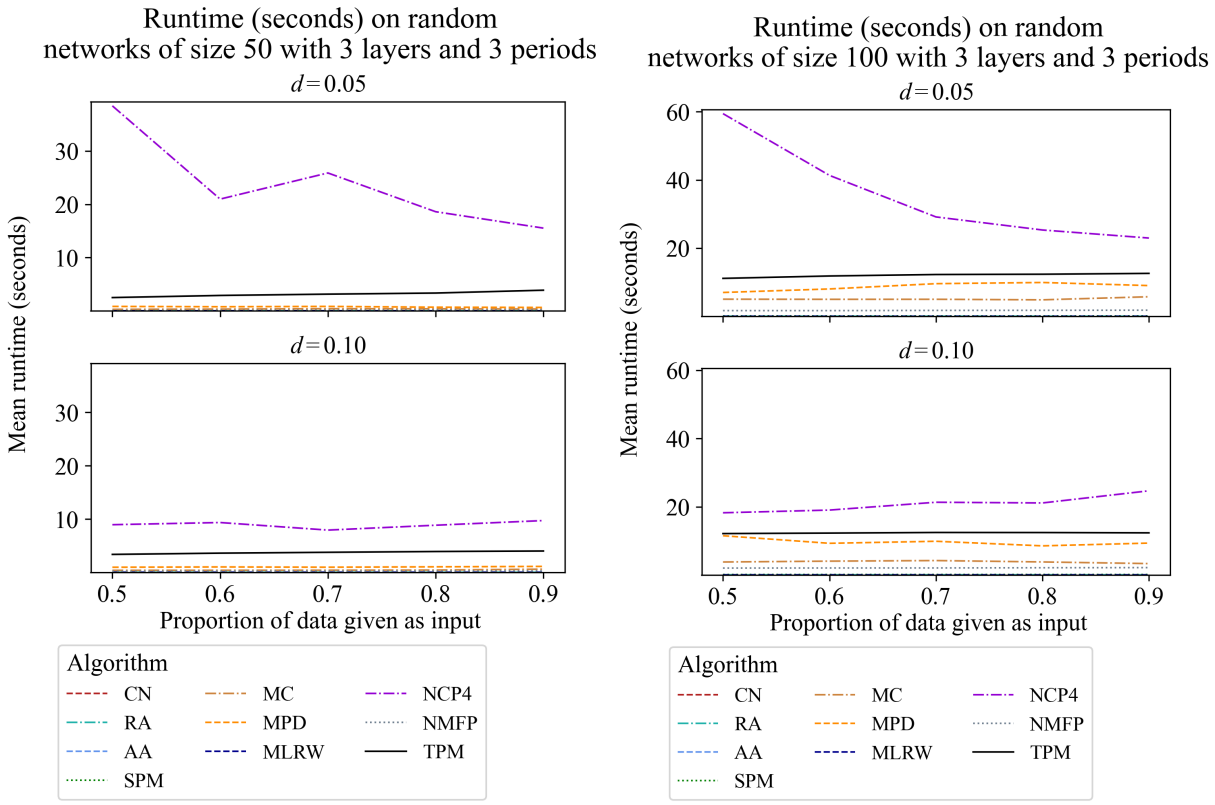
Figure D.22: Precision for inferring any missing links in dynamic multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

Figure D.23: Area under the receiver operating characteristic (AUROC) for inferring any missing links in dynamic multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

Figure D.24: Runtime for inferring any missing links in dynamic multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.

Appendix E

Covert Missing Link Inference Results

E.1 Multiplex Covert Networks

E.1.1 Belo Horizonte Co-Offenses

Figures E.1 and E.2 show the results for inferring missing links in the Belo Horizonte data [189].

E.1.2 Operazione Montagna Surveillance

Figures E.3 through E.5 show the results for inferring missing links in the Operazione Montagna data [45, 76].

E.1.3 Multiplex French Terror Network

Figures E.6 through E.8 show the results for inferring missing links in the multiplex network derived from the French terror tweets.

E.2 Dynamic Multiplex Covert Networks

E.2.1 Dynamic Multiplex French Terror Network

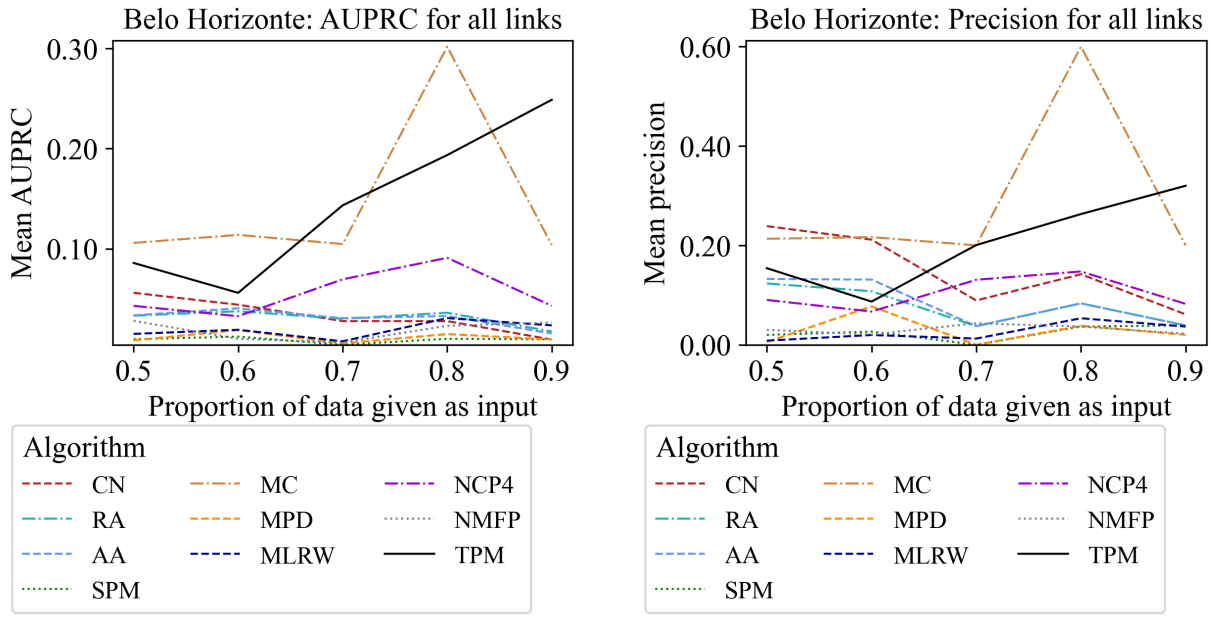
Figures E.9 through E.11 show the results for inferring missing links in the dynamic multiplex network derived from the French terror tweets.

E.2.2 ViNSAR Terrorist Rivalries

Figures E.12 through E.14 show the results for inferring missing links in the ViNSAR data set [45, 76].

E.2.3 Islamic Network Online Forum

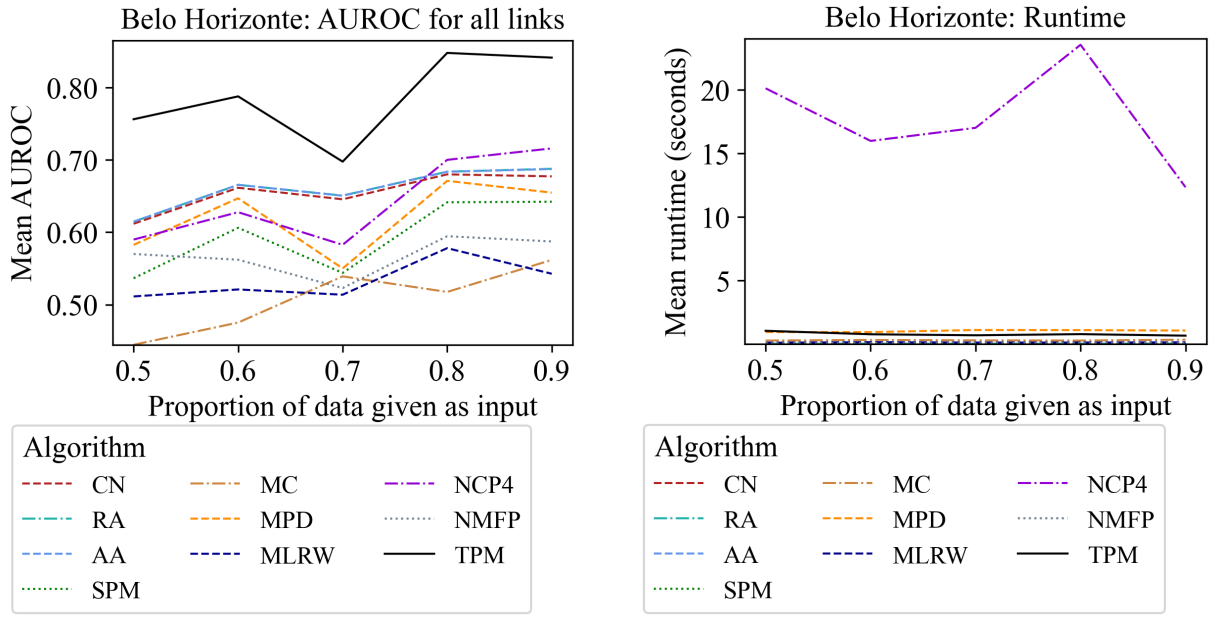
Figures E.15 through E.17 show the results for inferring missing links in the Islamic network online forum [197].



(a) AUPRC for inferring any missing links

(b) Precision for inferring any missing links

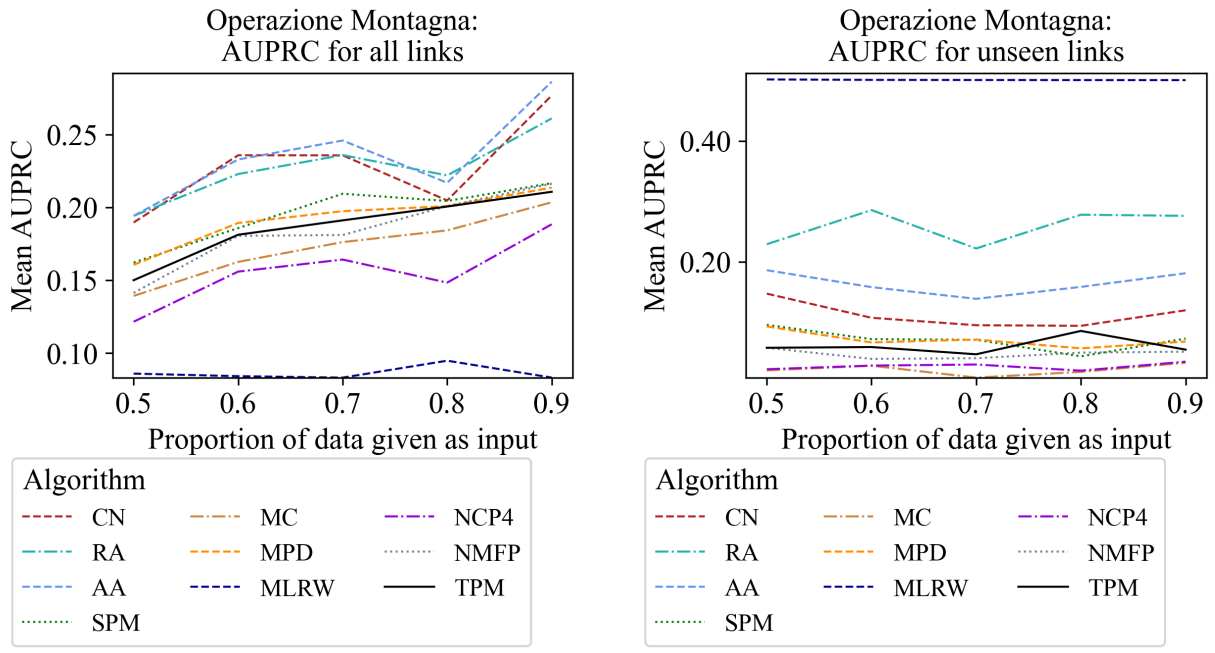
Figure E.1: Area under the precision-recall curve (AUPRC) and precision when inferring drug trafficking co-offenses in the Belo Horizonte data [189]. The horizontal axes show the proportions of total edge weight provided as input.



(a) AUROC for inferring any missing links

(b) Runtime (not counting cross-validation)

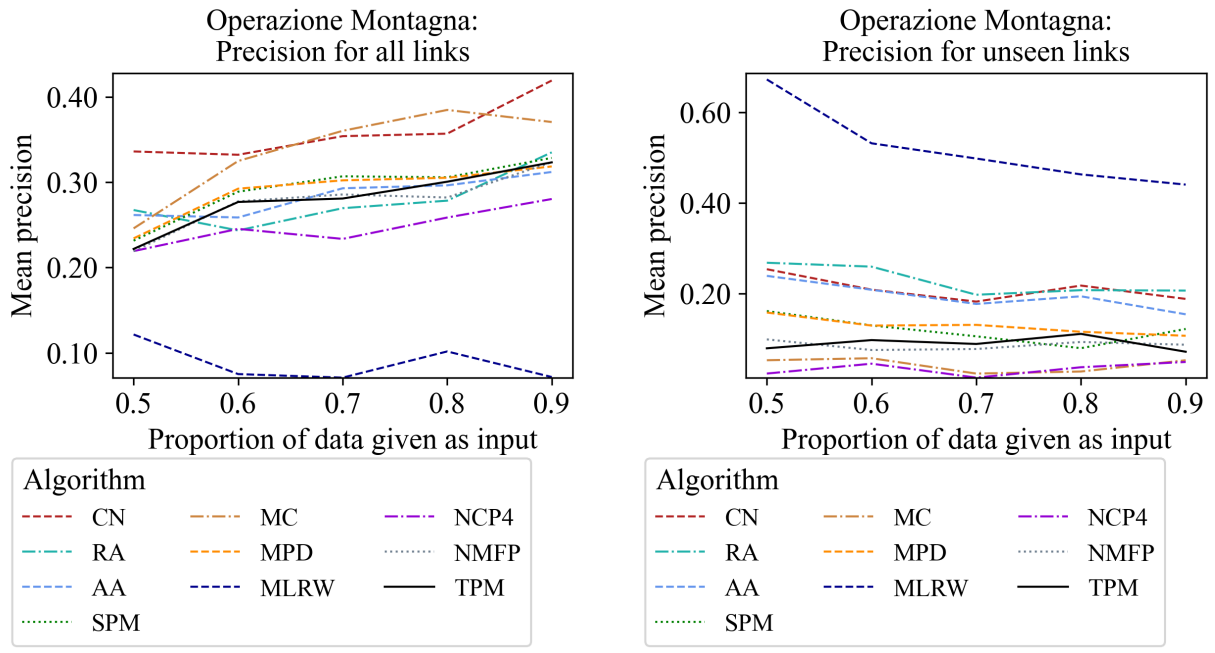
Figure E.2: Area under the receiver operating characteristic (AUROC) and runtime when inferring drug trafficking co-offenses in the Belo Horizonte data [189]. The horizontal axes show the proportions of total edge weight provided as input.



(a) AUPRC for inferring any missing links

(b) AUPRC for inferring wholly unseen links

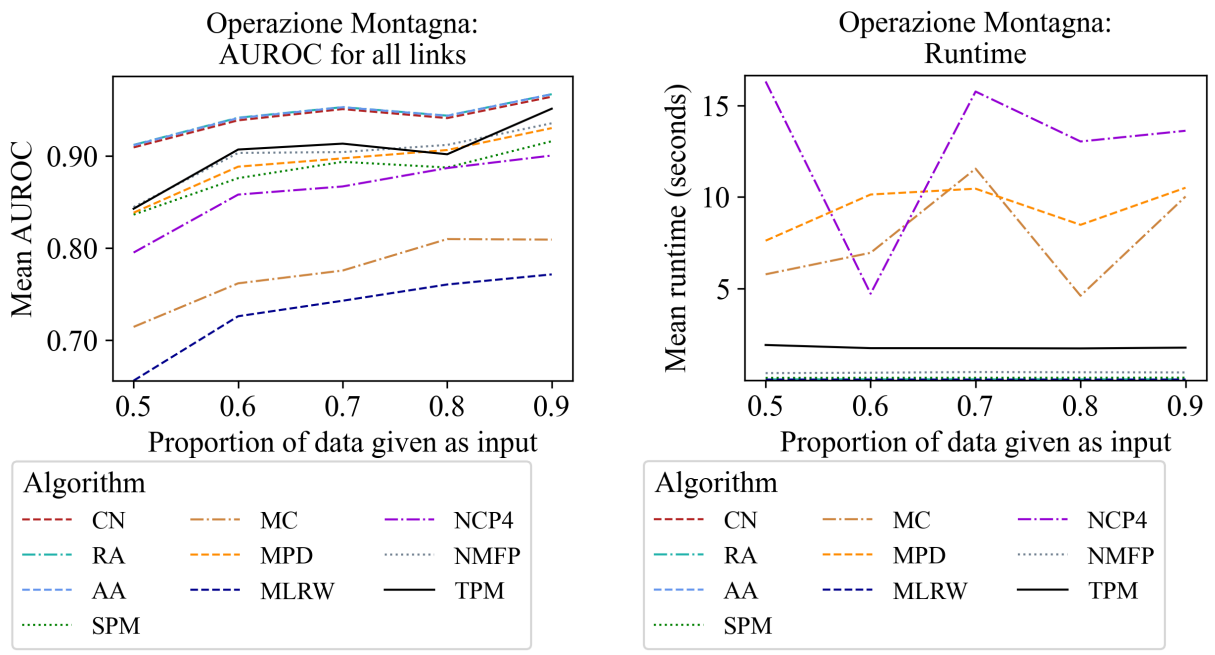
Figure E.3: Area under the precision-recall curve (AUPRC) when inferring in-person meetings between mafiosi in the Operazione Montagna data [45, 76]. The horizontal axes show the proportions of total edge weight provided as input.



(a) Precision for inferring any missing links

(b) Precision for inferring wholly unseen links

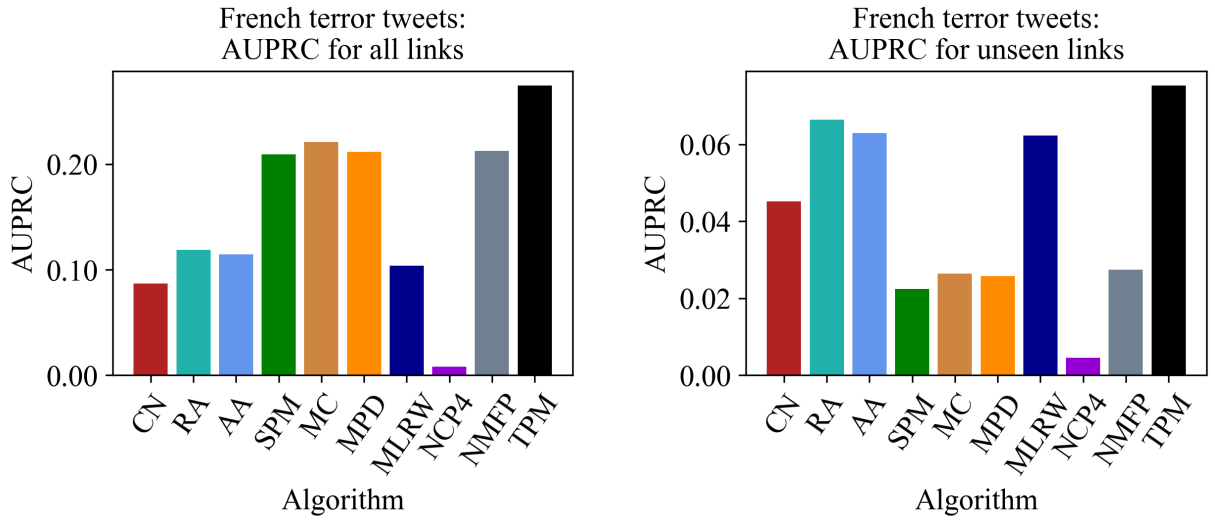
Figure E.4: Precision when inferring in-person meetings between mafiosi in the Operazione Montagna data [45, 76]. The horizontal axes show the proportions of total edge weight provided as input.



(a) AUROC for inferring any missing links

(b) Runtime (not counting cross-validation)

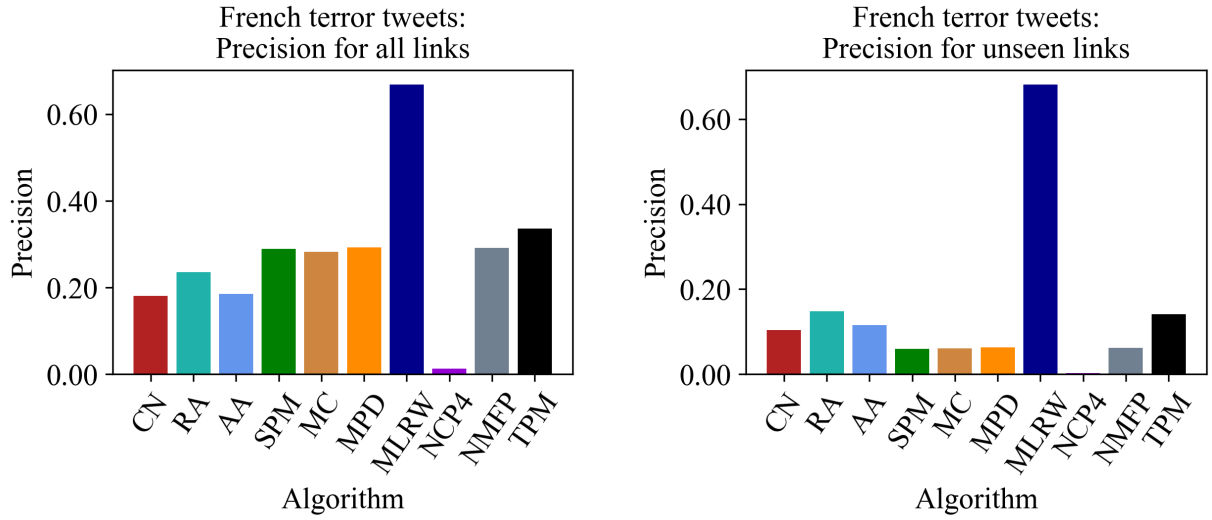
Figure E.5: Area under the receiver operating characteristic (AUROC) and runtime when inferring in-person meetings between mafiosi in the Operazione Montagna data [45, 76]. The horizontal axes show the proportions of total edge weight provided as input.



(a) AUPRC for inferring any missing links

(b) AUPRC for inferring wholly unseen links

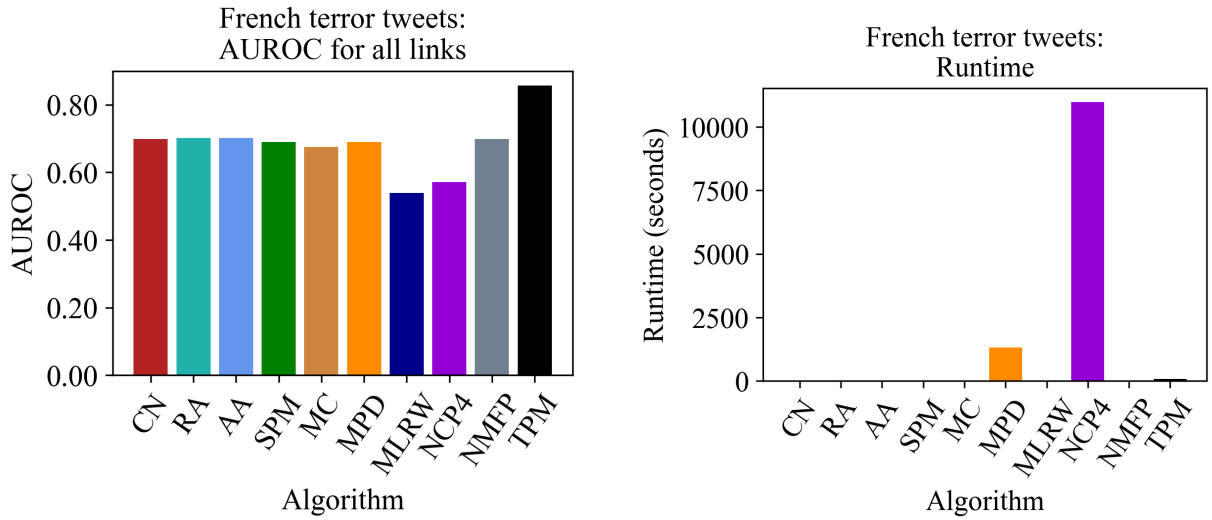
Figure E.6: Area under the precision-recall curve (AUPRC) when inferring communication between Twitter users discussing the 2020 French terror attacks.



(a) Precision for inferring any missing links

(b) Precision for inferring wholly unseen links

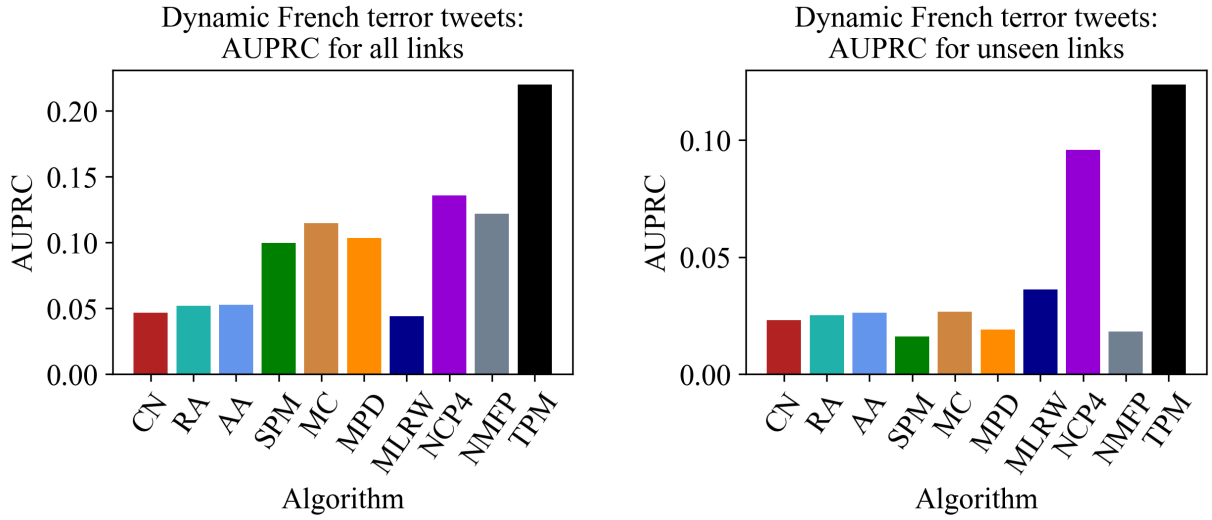
Figure E.7: Precision when inferring communication between Twitter users discussing the 2020 French terror attacks.



(a) AUROC for inferring any missing links

(b) Runtime (not counting cross-validation)

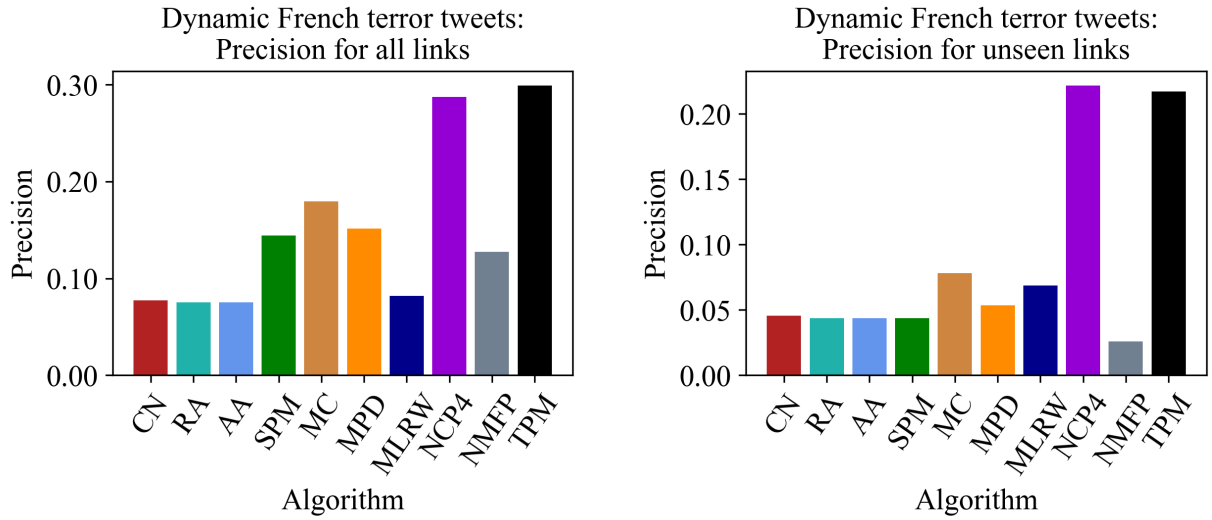
Figure E.8: Area under the receiver operating characteristic (AUROC) and runtime when inferring communication between Twitter users discussing the 2020 French terror attacks.



(a) AUPRC for inferring any missing links

(b) AUPRC for inferring wholly unseen links

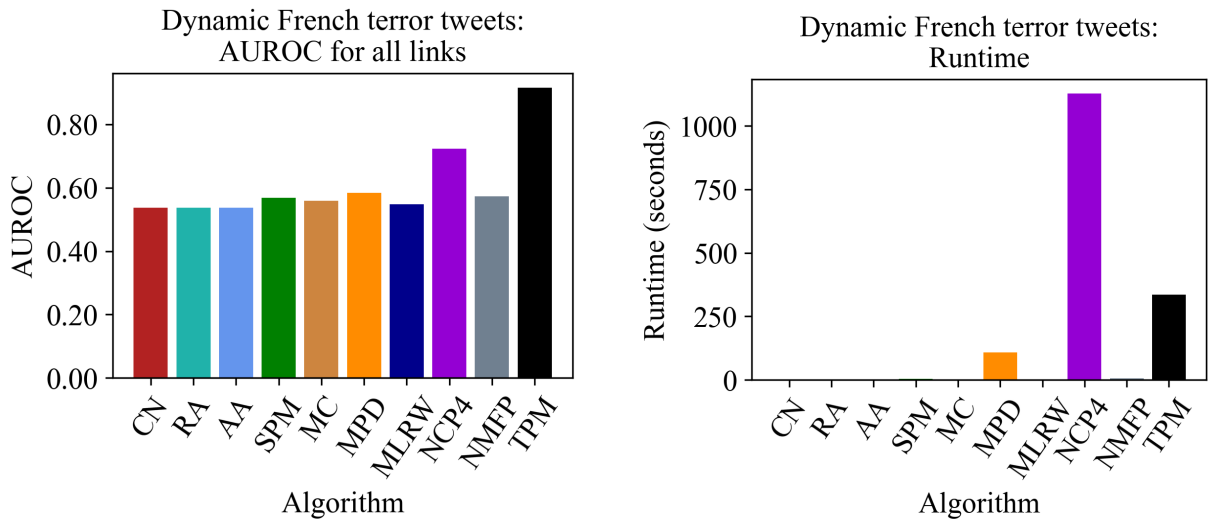
Figure E.9: Area under the precision-recall curve (AUPRC) when inferring reply and mention connections between Twitter users discussing the 2020 French terror attacks.



(a) Precision for inferring any missing links

(b) Precision for inferring wholly unseen links

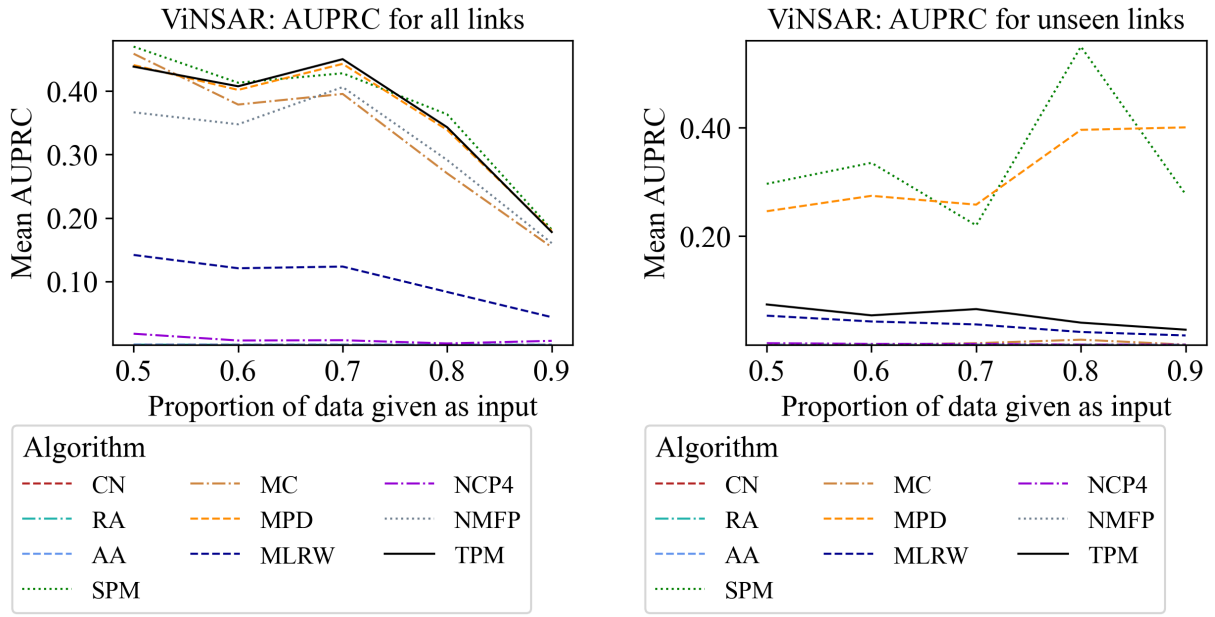
Figure E.10: Precision when inferring reply and mention connections between Twitter users discussing the 2020 French terror attacks.



(a) AUROC for inferring any missing links

(b) Runtime (not counting cross-validation)

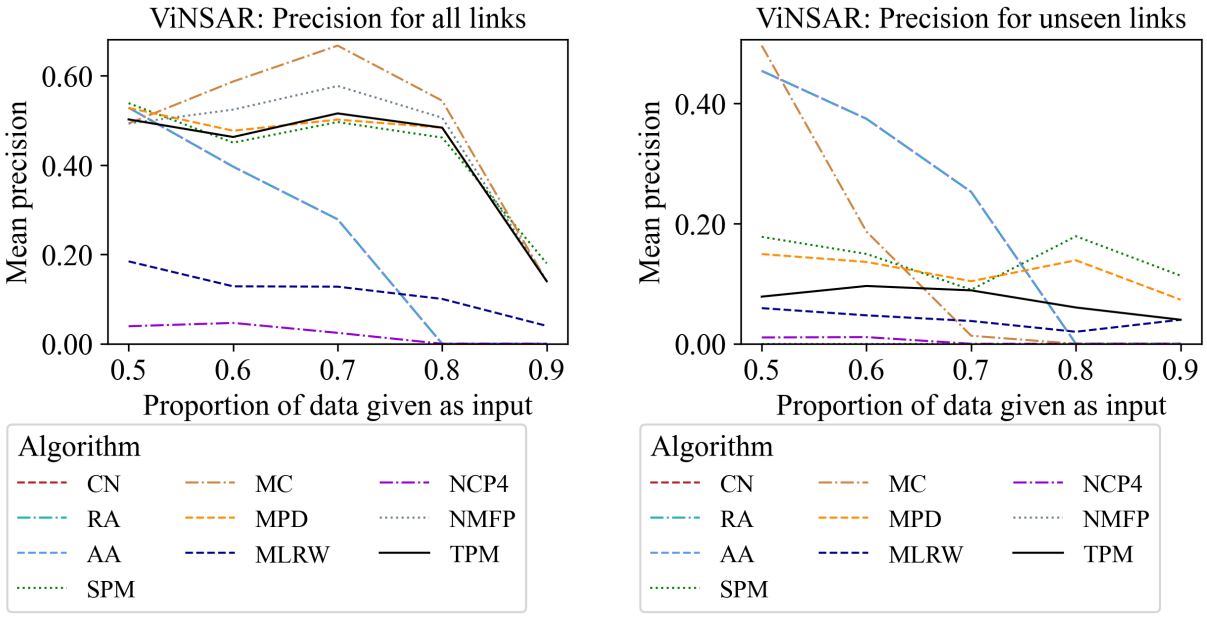
Figure E.11: Area under the receiver operating characteristic (AUROC) and runtime when inferring reply and mention connections between Twitter users discussing the 2020 French terror attacks.



(a) AUPRC for inferring any missing links

(b) AUPRC for inferring wholly unseen links

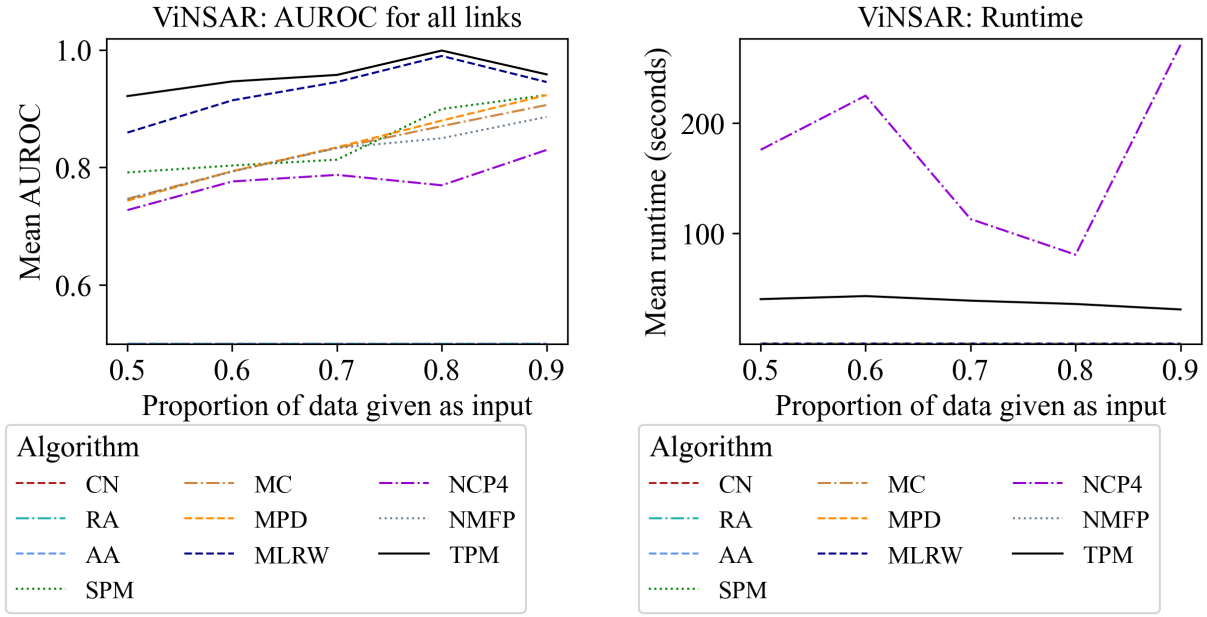
Figure E.12: Area under the precision-recall curve (AUPRC) when inferring deadly rivalries between terrorist groups in the ViNSAR data set [51]. The horizontal axes show the proportions of total edge weight provided as input.



(a) Precision for inferring any missing links

(b) Precision for inferring wholly unseen links

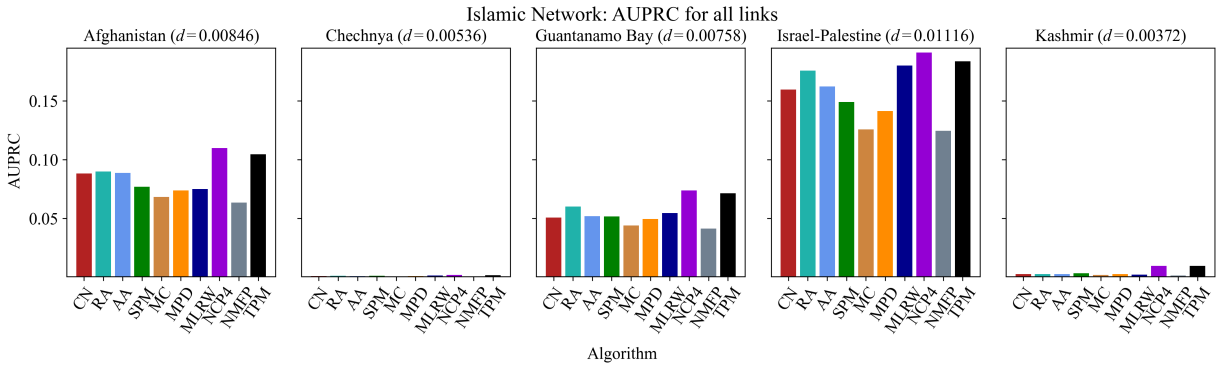
Figure E.13: Precision when inferring deadly rivalries between terrorist groups in the ViNSAR data set [51]. The horizontal axes show the proportions of total edge weight provided as input.



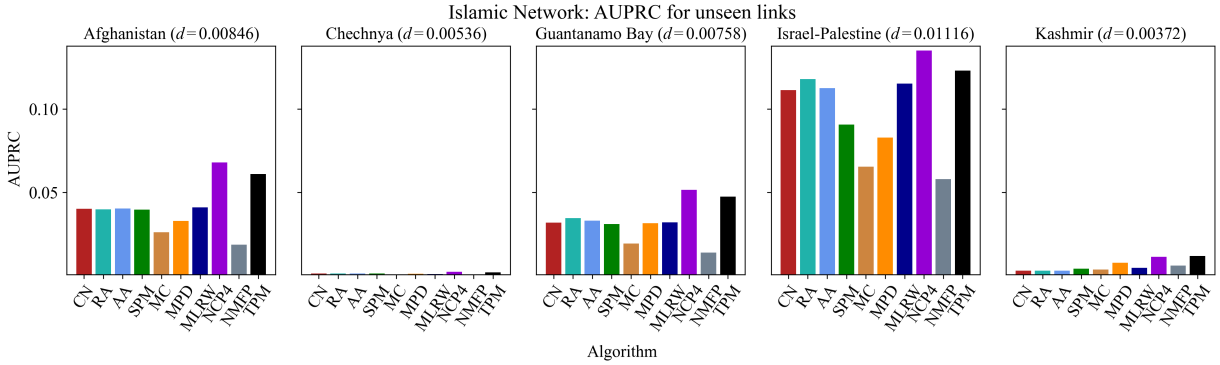
(a) AUROC for inferring any missing links

(b) Runtime (not counting cross-validation)

Figure E.14: Area under the receiver operating characteristic (AUROC) and runtime when inferring deadly rivalries between terrorist groups in the ViNSAR data set [51]. The horizontal axes show the proportions of total edge weight provided as input.

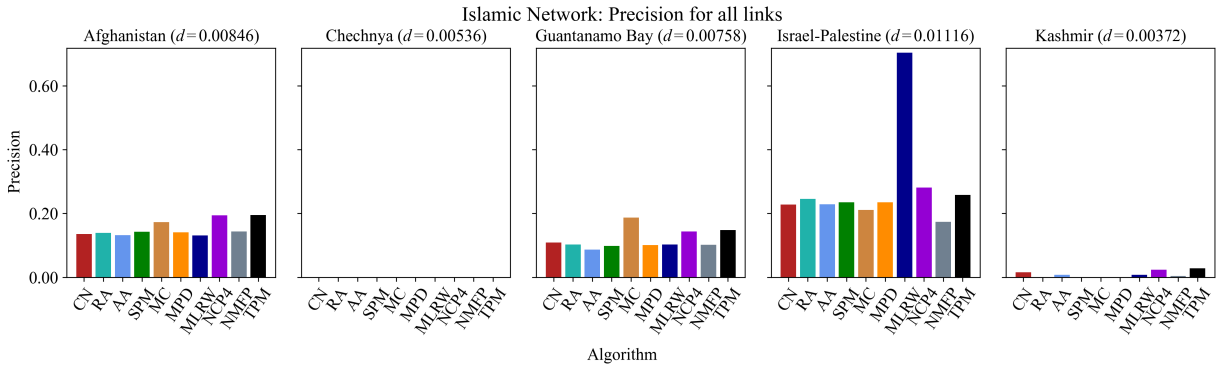


(a) AUPRC for inferring any missing links

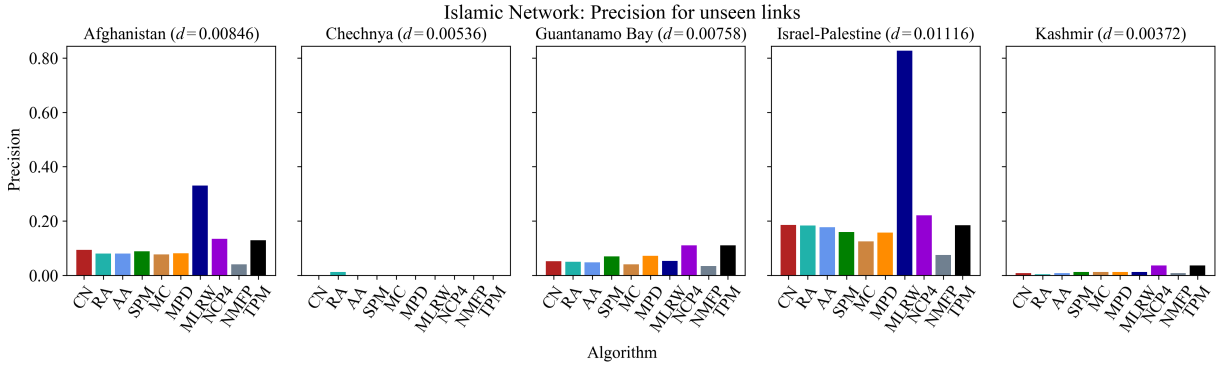


(b) AUPRC for inferring wholly unseen links

Figure E.15: Area under the precision-recall curve (AUPRC) when inferring topic-specific communication in the Islamic Network online forum [197]. Each subplot shows the performance for inferring communication on that topic when provided with the communication related to the other four topics. Each subplot title also indicates the density d of the communication network for that topic.

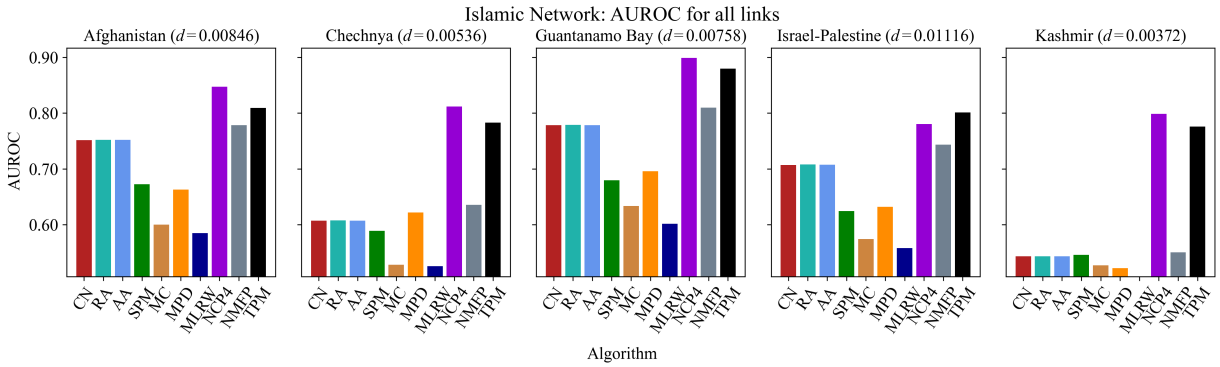


(a) Precision for inferring any missing links

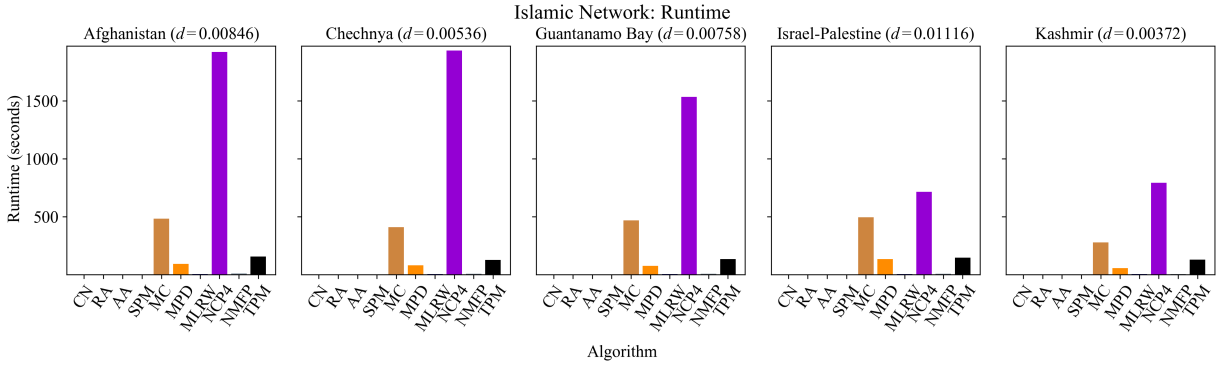


(b) Precision for inferring wholly unseen links

Figure E.16: Precision when inferring topic-specific communication in the Islamic Network online forum [197]. Each subplot shows the performance for inferring communication on that topic when provided with the communication related to the other four topics. Each subplot title also indicates the density d of the communication network for that topic.



(a) AUROC for inferring any missing links



(b) Runtime (not counting cross-validation)

Figure E.17: Area under the receiver operating characteristic (AUROC) and runtime when inferring topic-specific communication in the Islamic Network online forum [197]. Each subplot shows the performance for inferring communication on that topic when provided with the communication related to the other four topics. Each subplot title also indicates the density d of the communication network for that topic.

Appendix F

Simulated Spurious Link Identification Results

F.1 Multiplex Stylized Networks

F.1.1 Small-World Networks

Figures F.1 through F.4 show the results for inferring missing links in multiplex small-world networks.

F.1.2 Scale-Free Networks

Figures F.5 through F.8 show the results for inferring missing links in multiplex scale-free networks.

F.1.3 Uniformly Random Networks

Figures F.9 through F.12 show the results for inferring missing links in multiplex uniformly random networks.

F.2 Dynamic Multiplex Stylized Networks

F.2.1 Small-World Networks

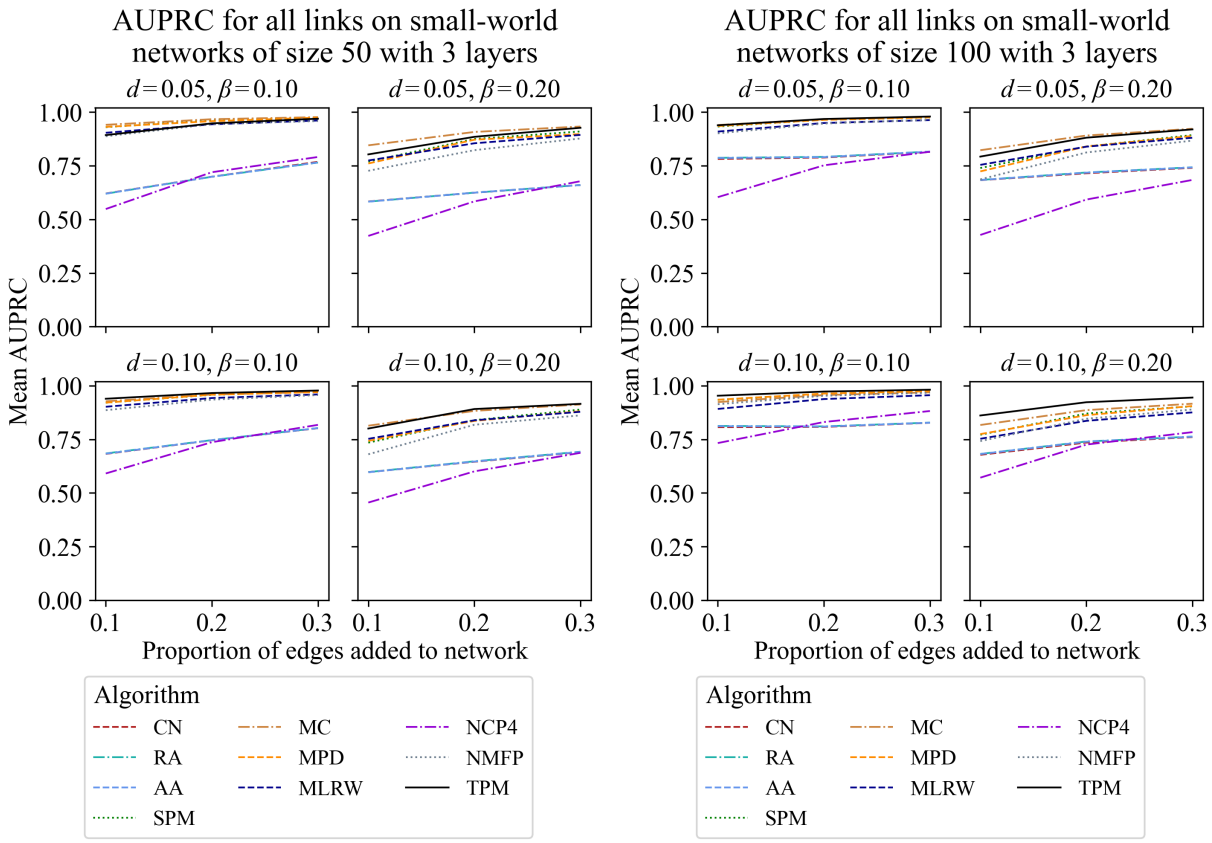
Figures F.13 through F.16 show the results for inferring missing links in dynamic multiplex small-world networks.

F.2.2 Scale-Free Networks

Figures F.17 through F.20 show the results for inferring missing links in dynamic multiplex scale-free networks.

F.2.3 Uniformly Random Networks

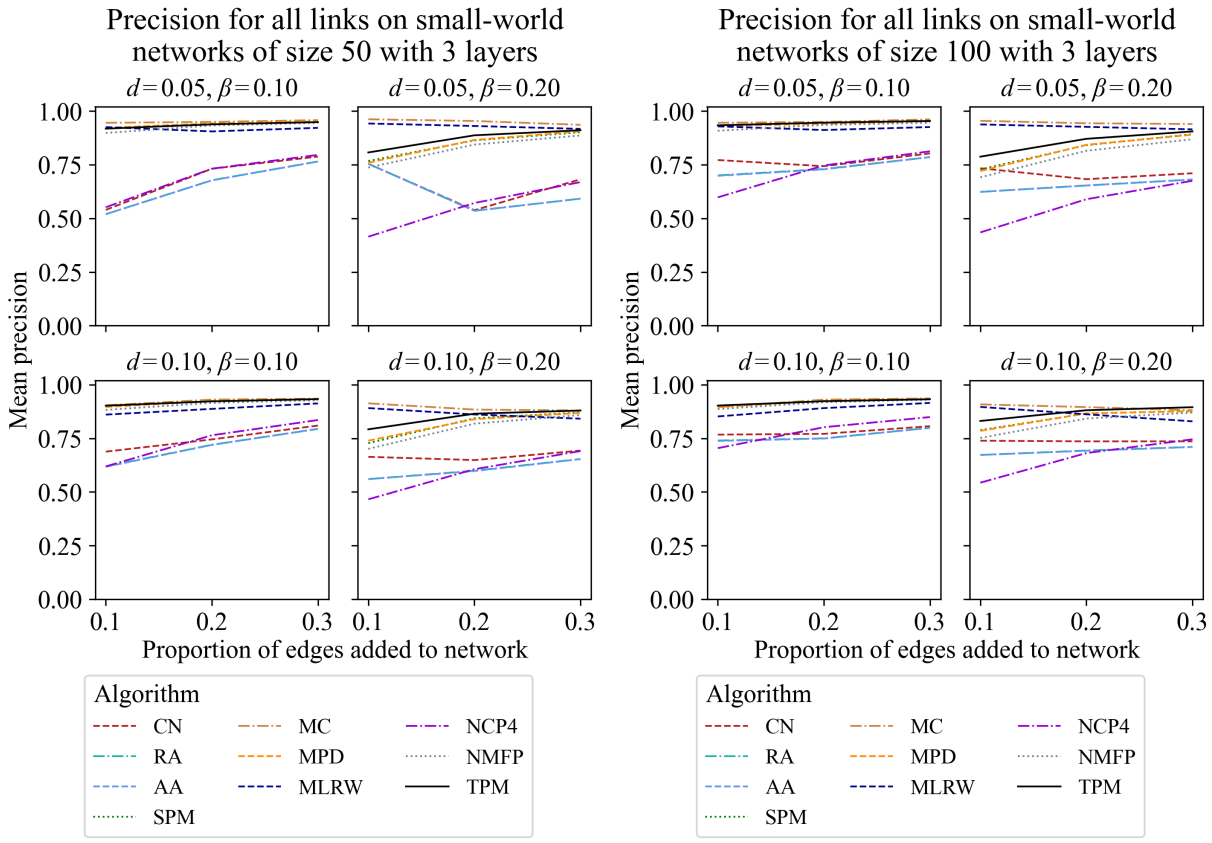
Figures F.21 through F.24 show the results for inferring missing links in dynamic multiplex uniformly random networks.



(a) Fifty-node networks

(b) Hundred-node networks

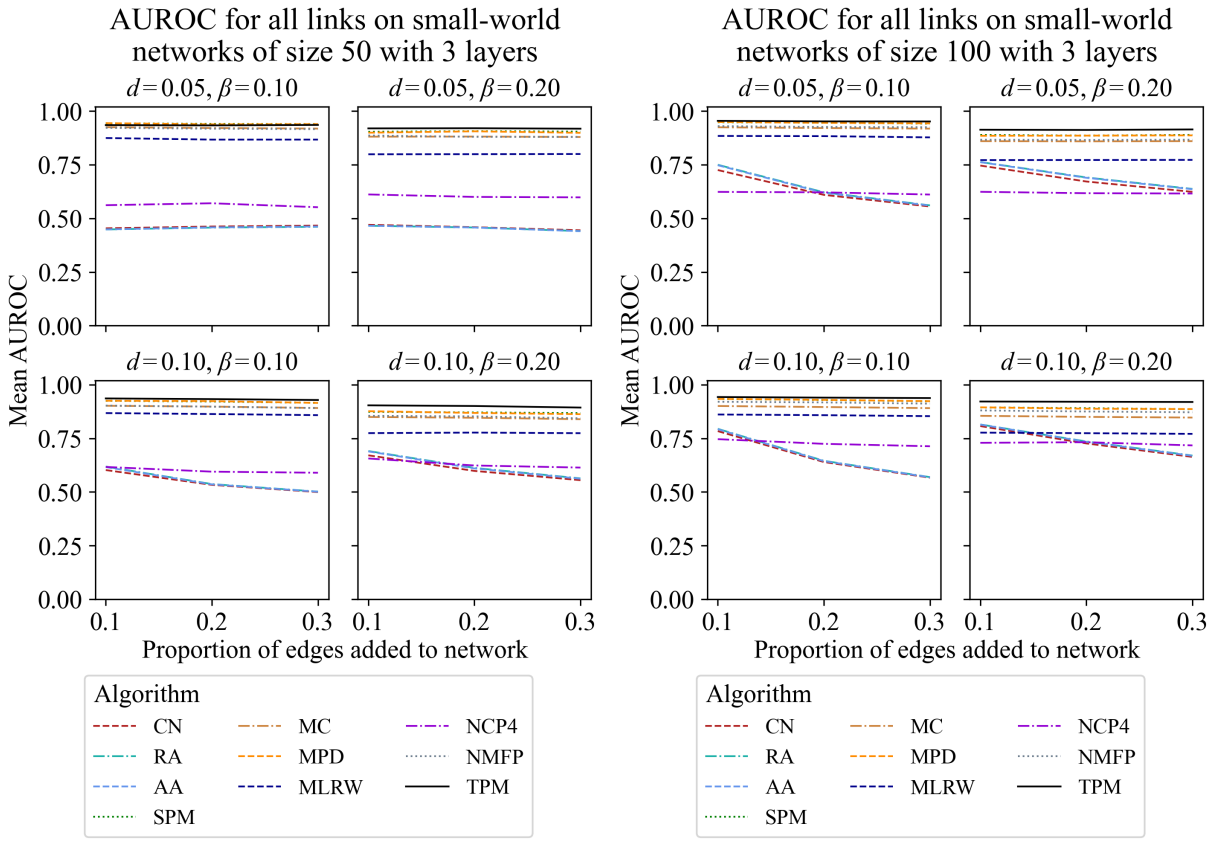
Figure F.1: Area under the precision-recall curve (AUPRC) for identifying any spurious links in multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

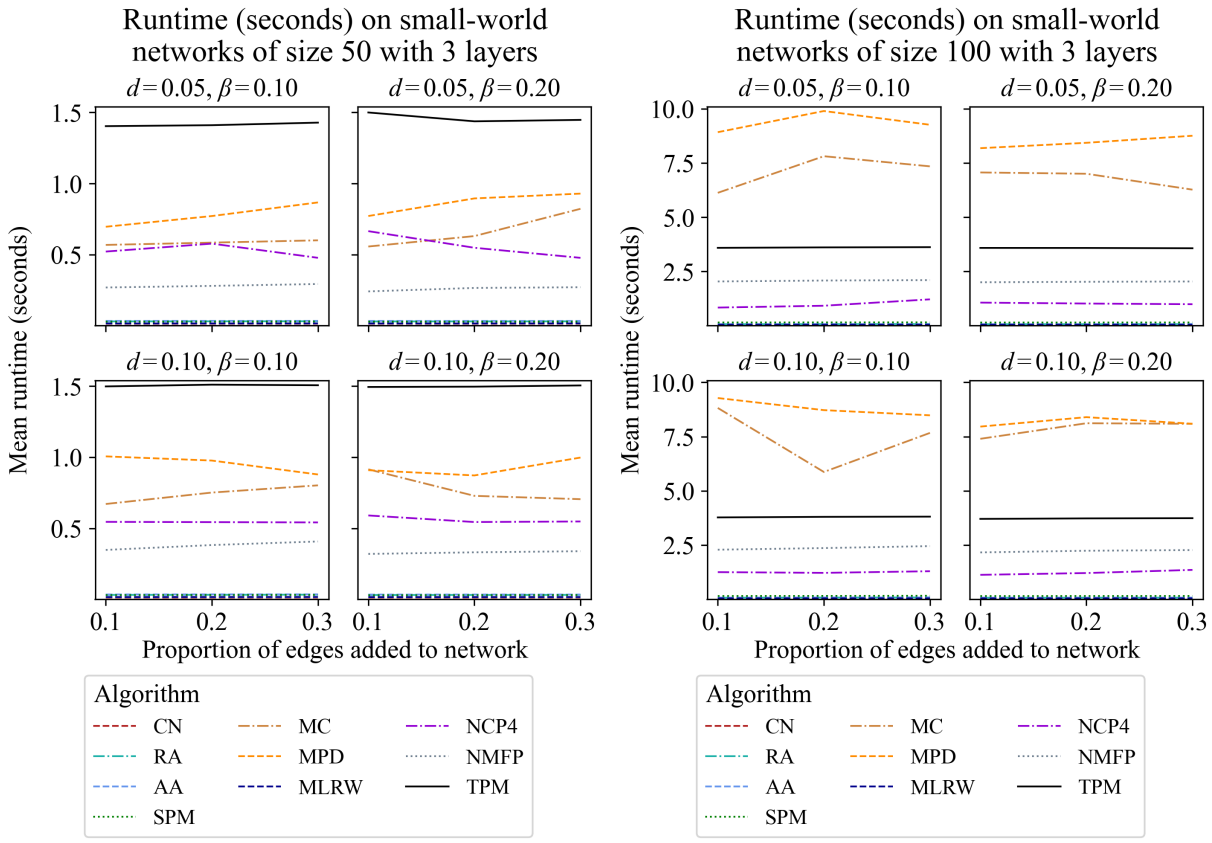
Figure F.2: Precision for identifying any spurious links in multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

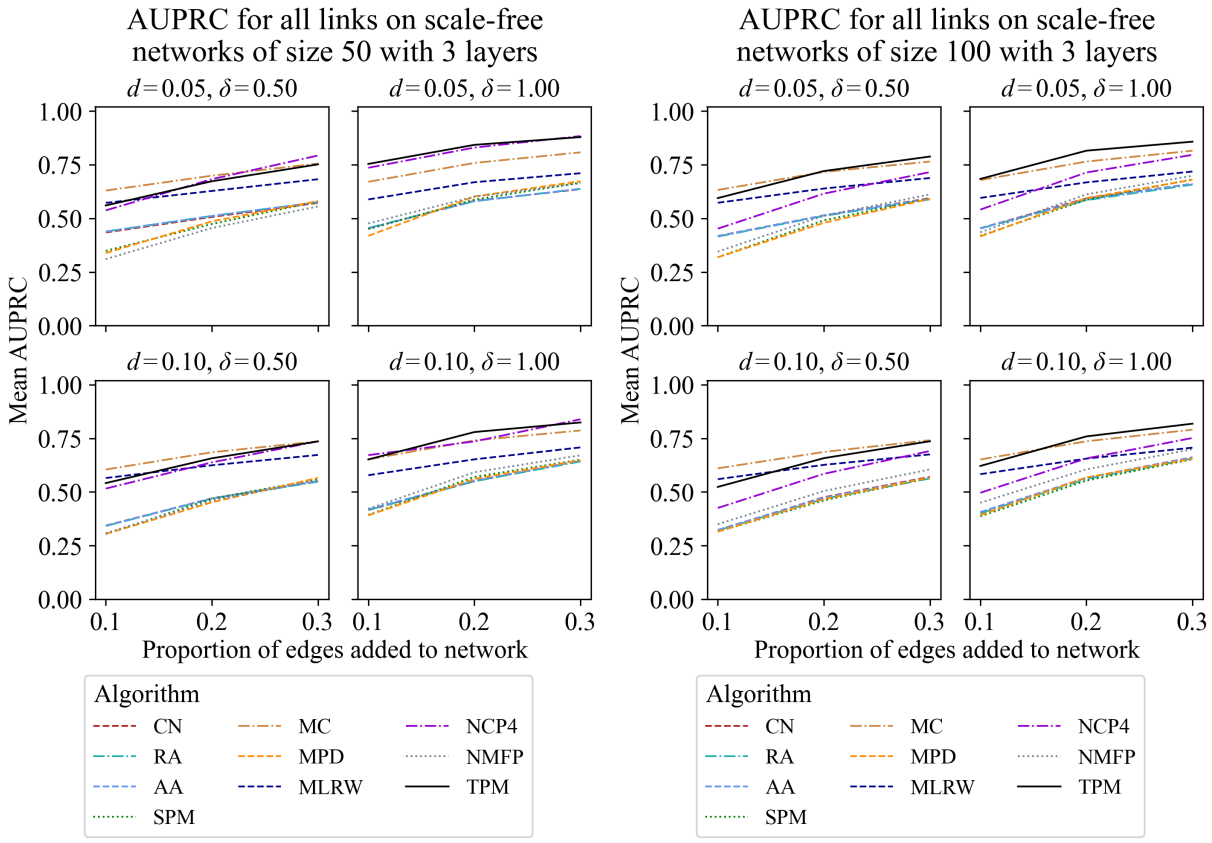
Figure F.3: Area under the receiver operating characteristic (AUROC) for identifying any spurious links in multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

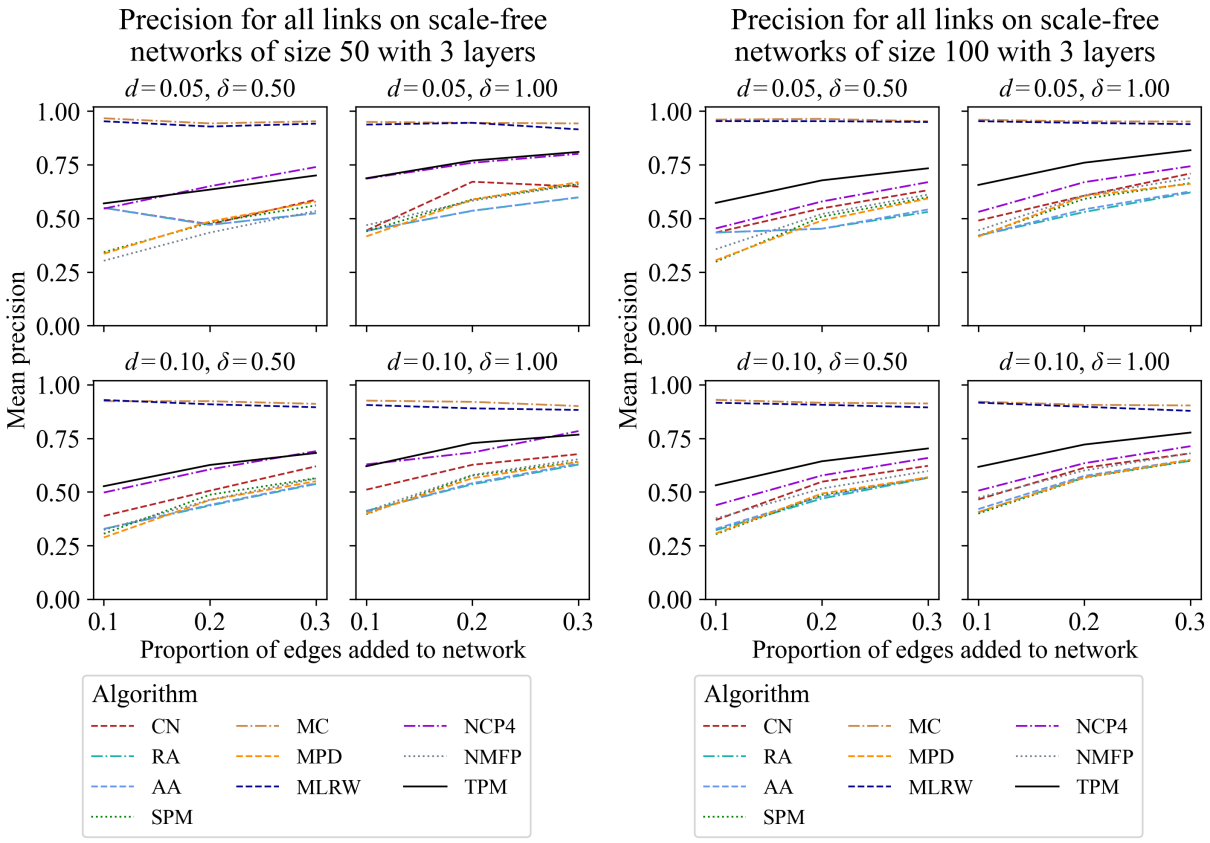
Figure F.4: Runtime for identifying any spurious links in multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

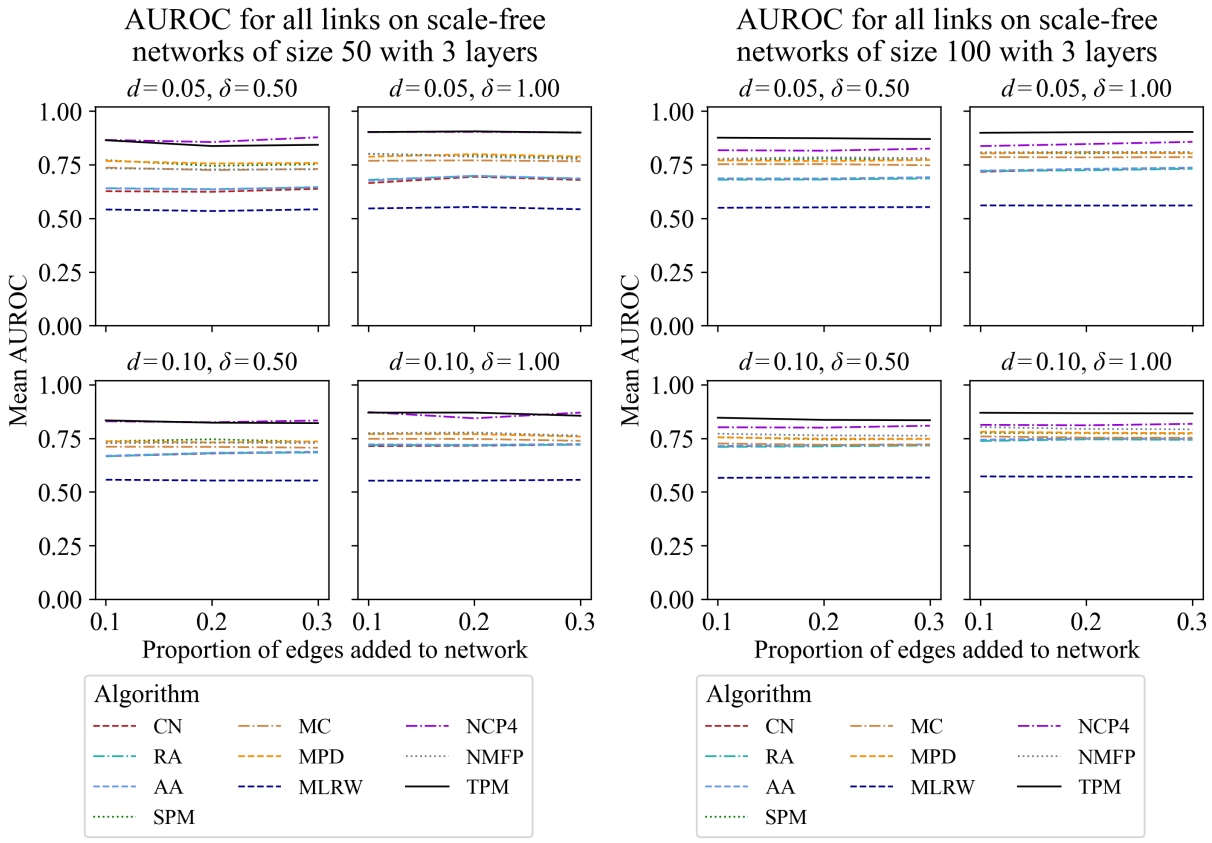
Figure F.5: Area under the precision-recall curve (AUPRC) for identifying any spurious links in multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

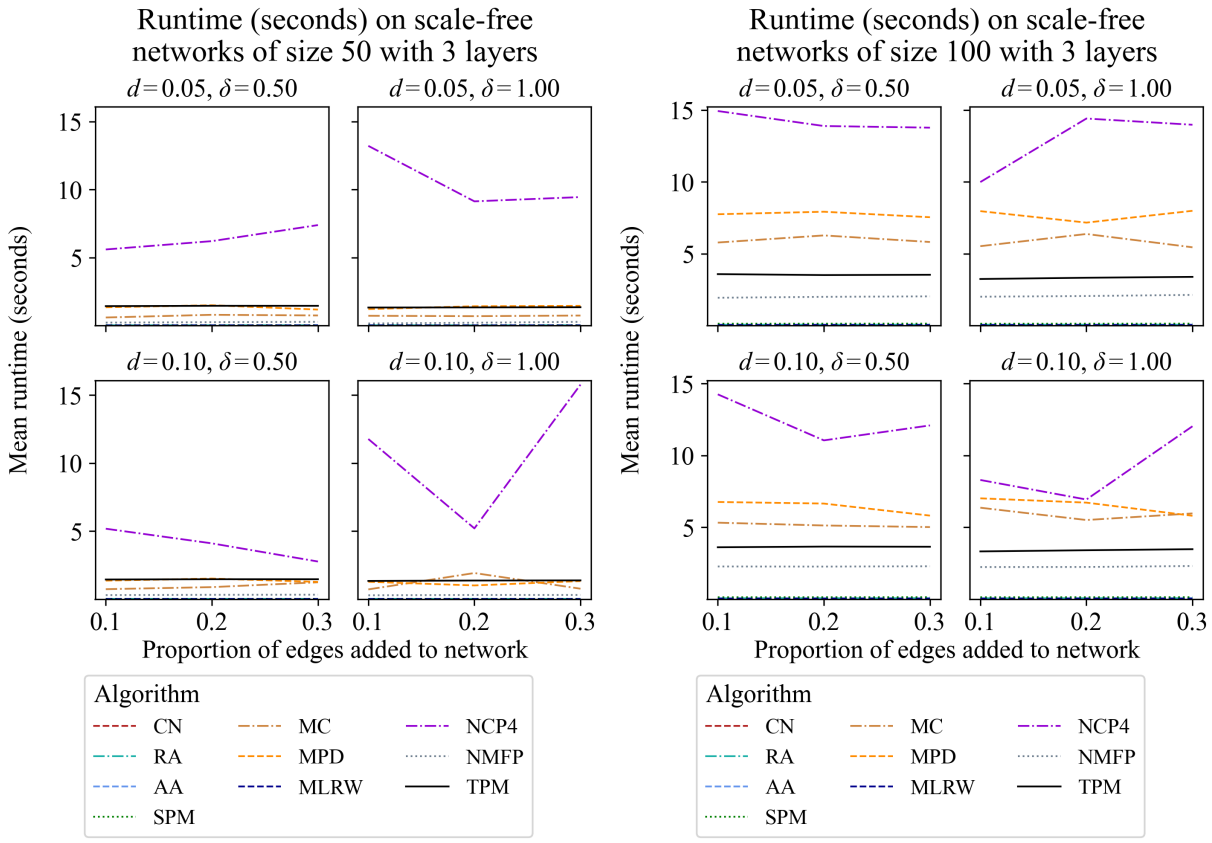
Figure F.6: Precision for identifying any spurious links in multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

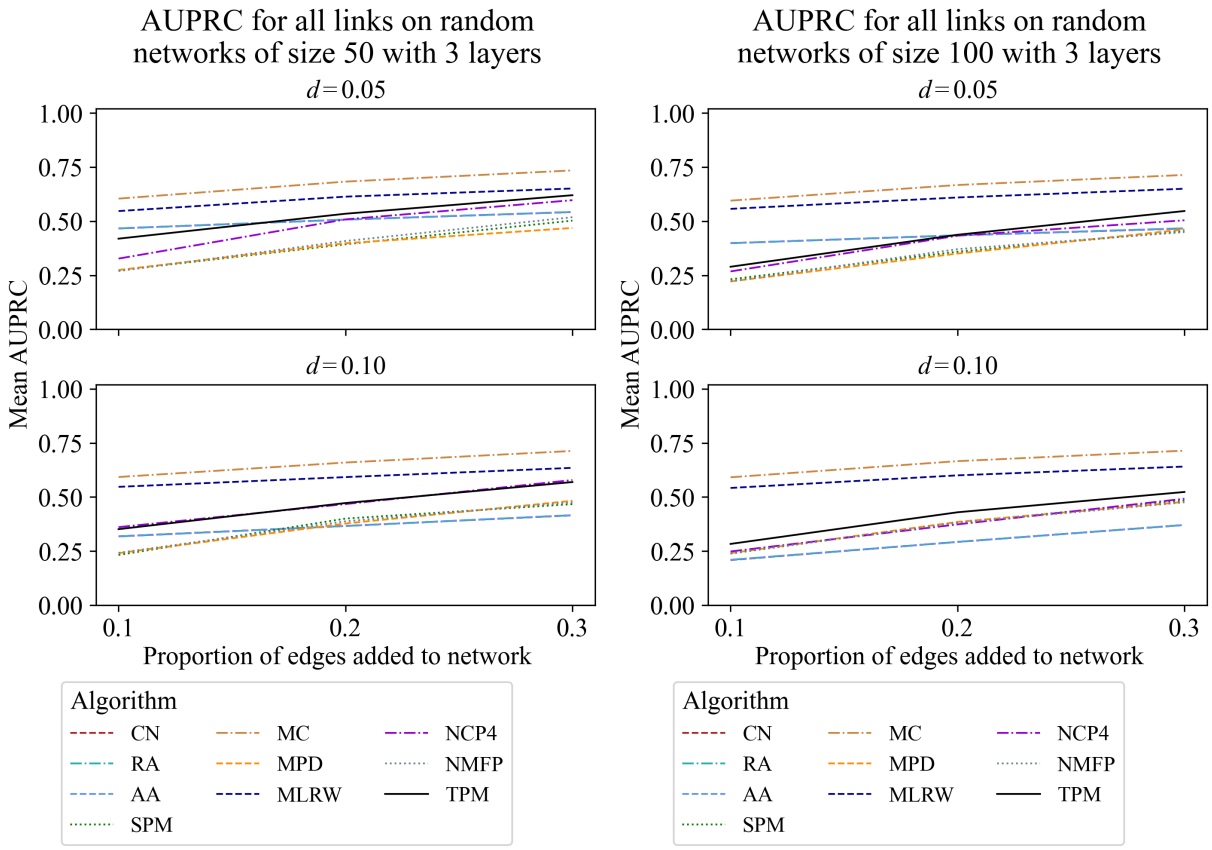
Figure F.7: Area under the receiver operating characteristic (AUROC) for identifying any spurious links in multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

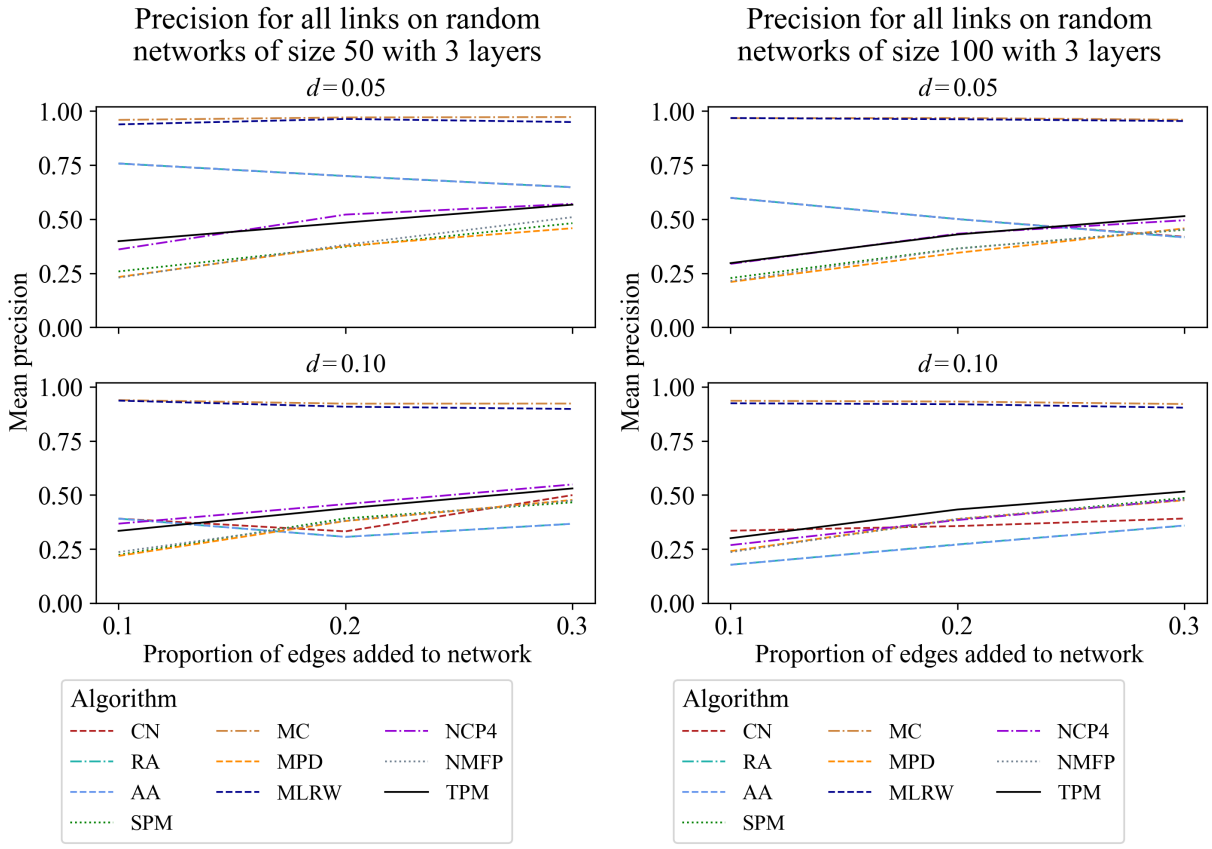
Figure F.8: Runtime for identifying any spurious links in multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

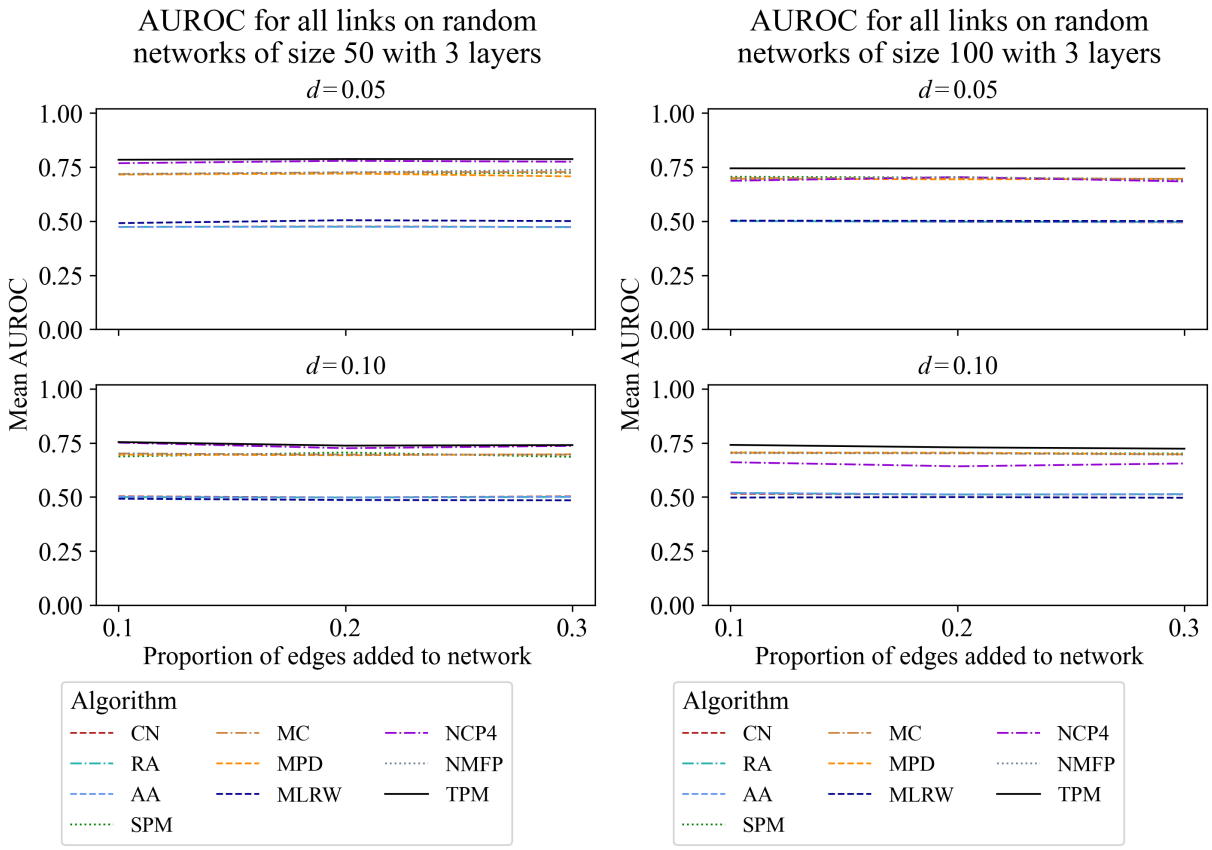
Figure F.9: Area under the precision-recall curve (AUPRC) for identifying any spurious links in multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

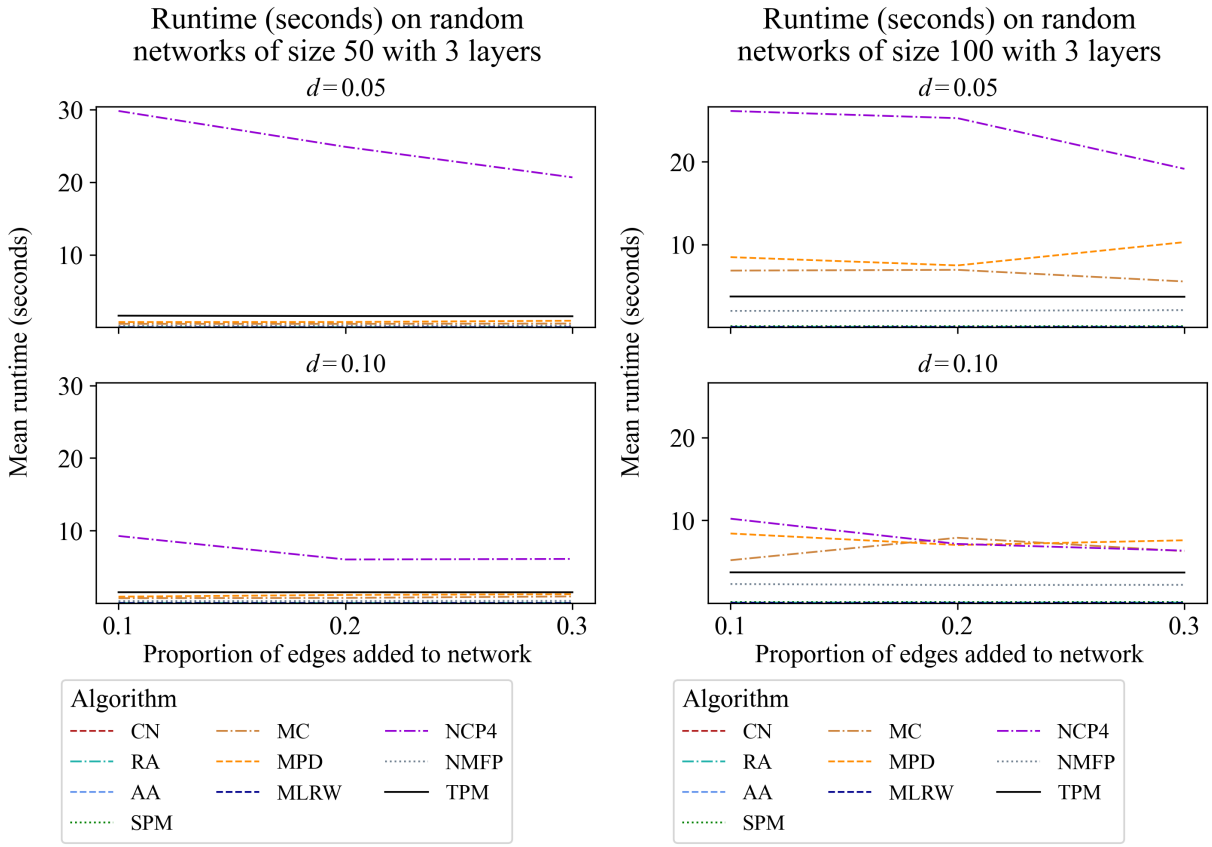
Figure F.10: Precision for identifying any spurious links in multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

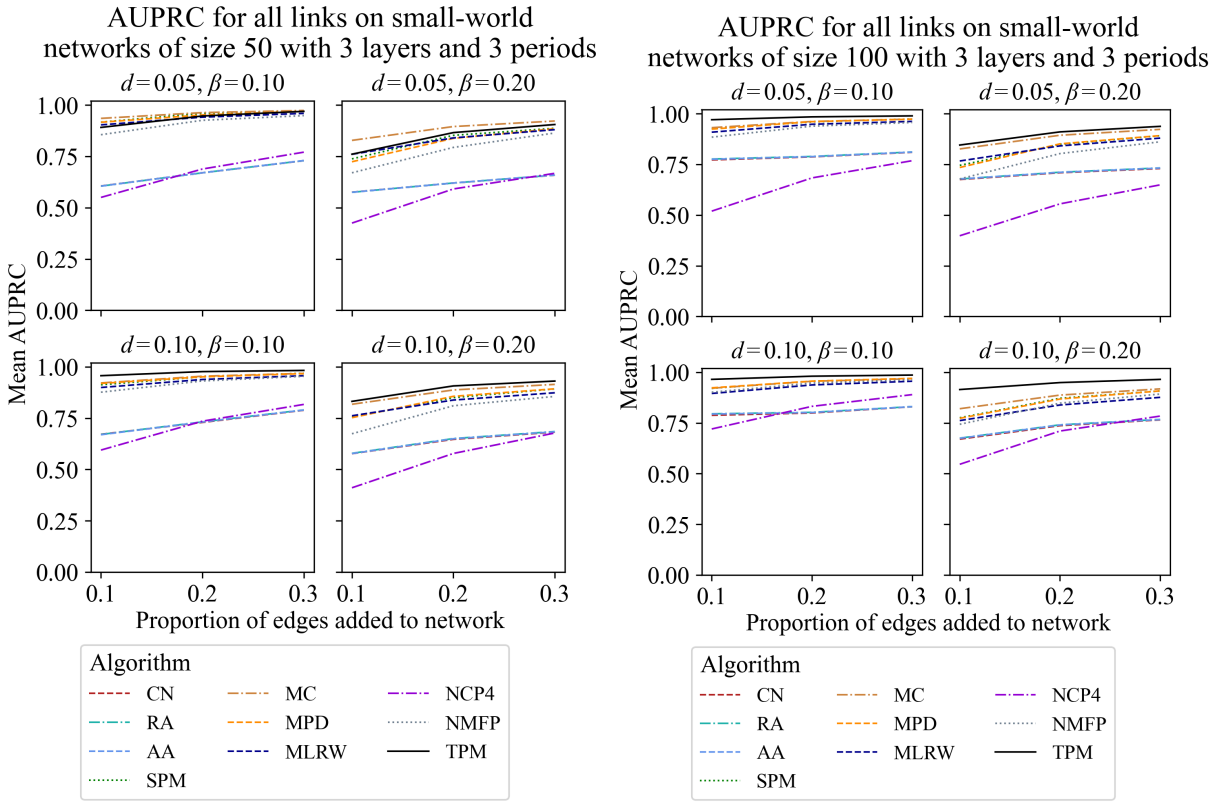
Figure F.11: Area under the receiver operating characteristic (AUROC) for identifying any spurious links in multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

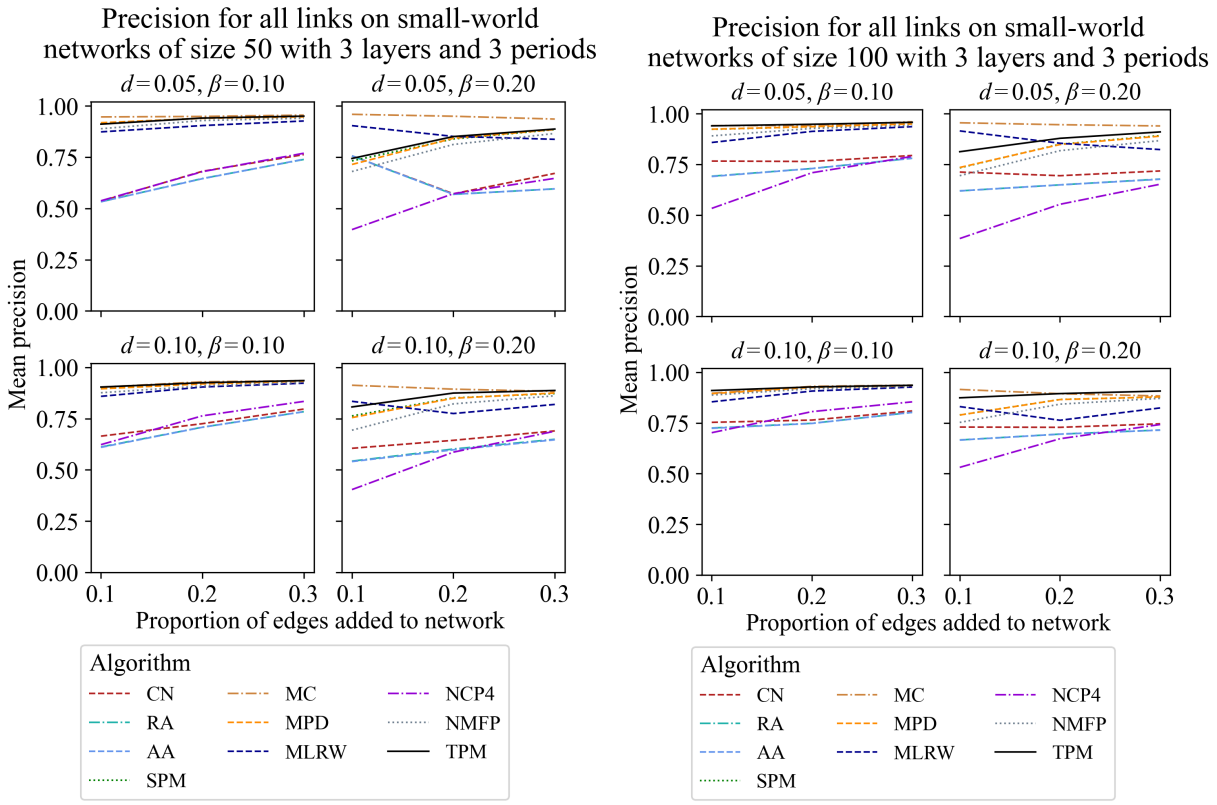
Figure F.12: Runtime for identifying any spurious links in multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

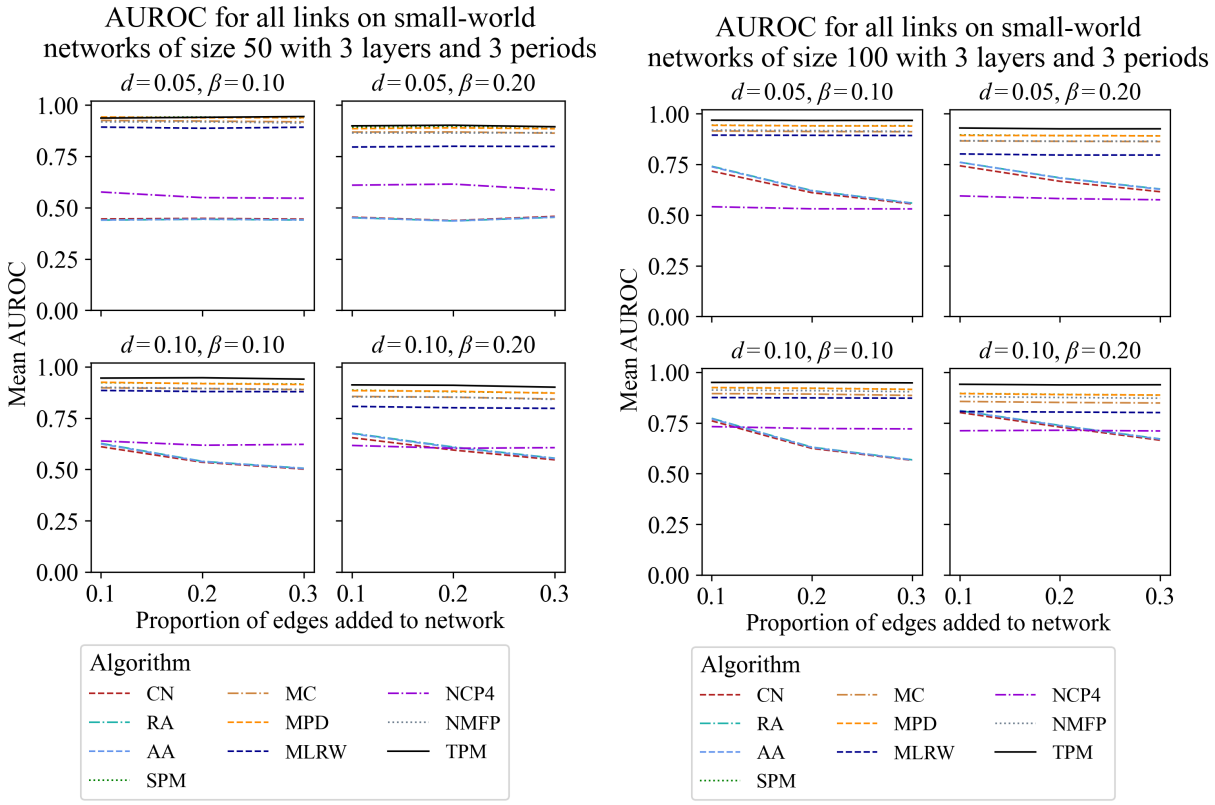
Figure F.13: Area under the precision-recall curve (AUPRC) for identifying any spurious links in dynamic multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

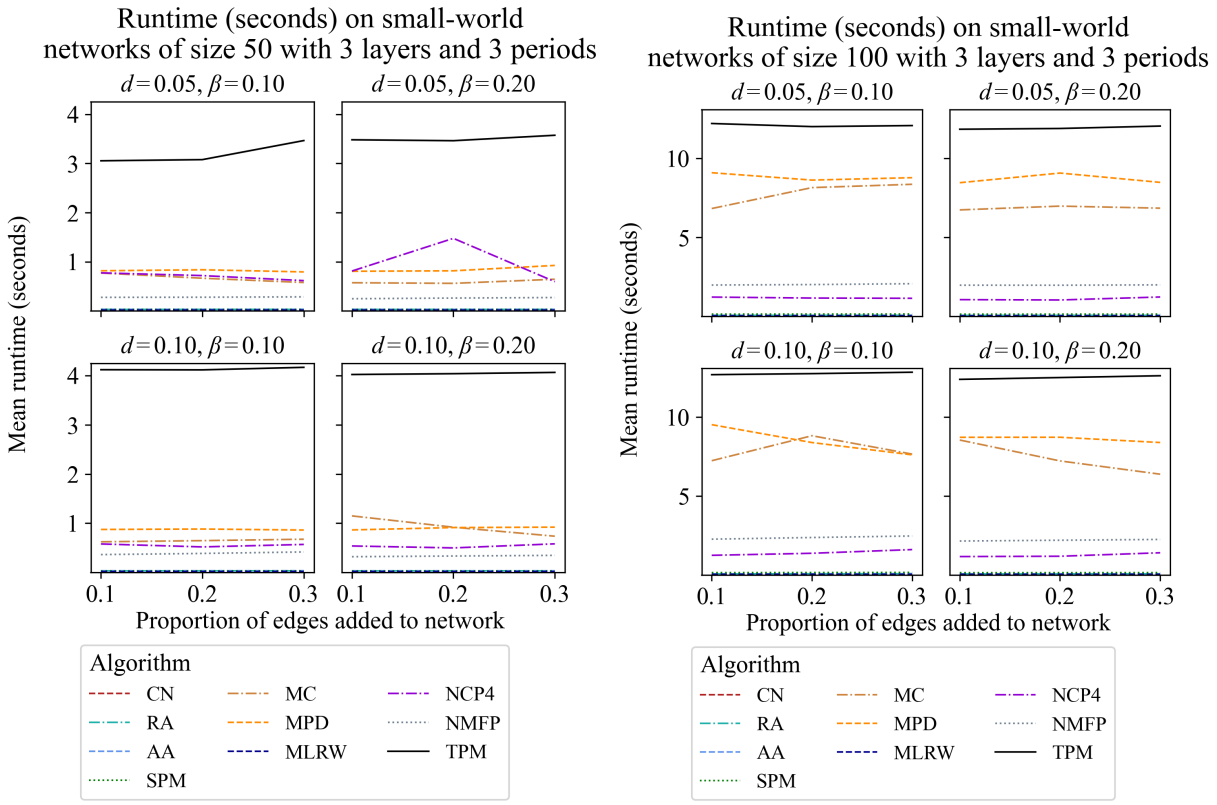
Figure F.14: Precision for identifying any spurious links in dynamic multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

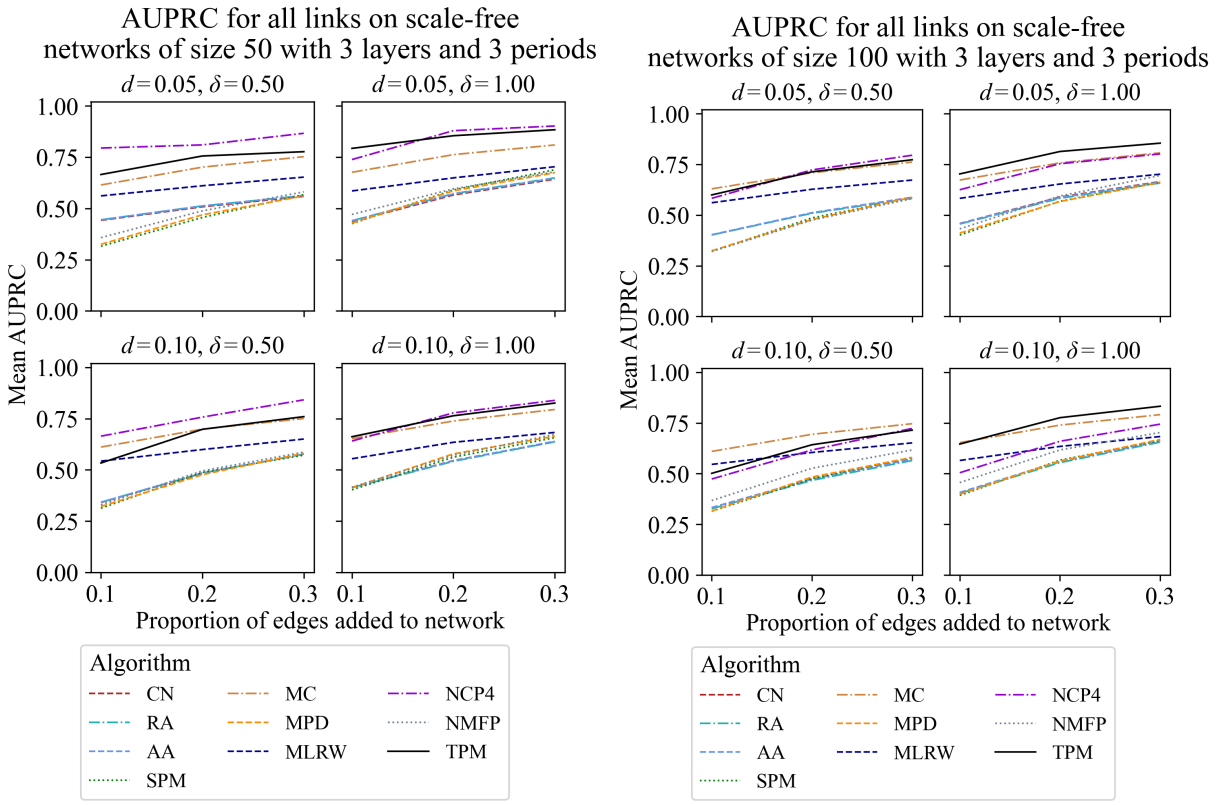
Figure F.15: Area under the receiver operating characteristic (AUROC) for identifying any spurious links in dynamic multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

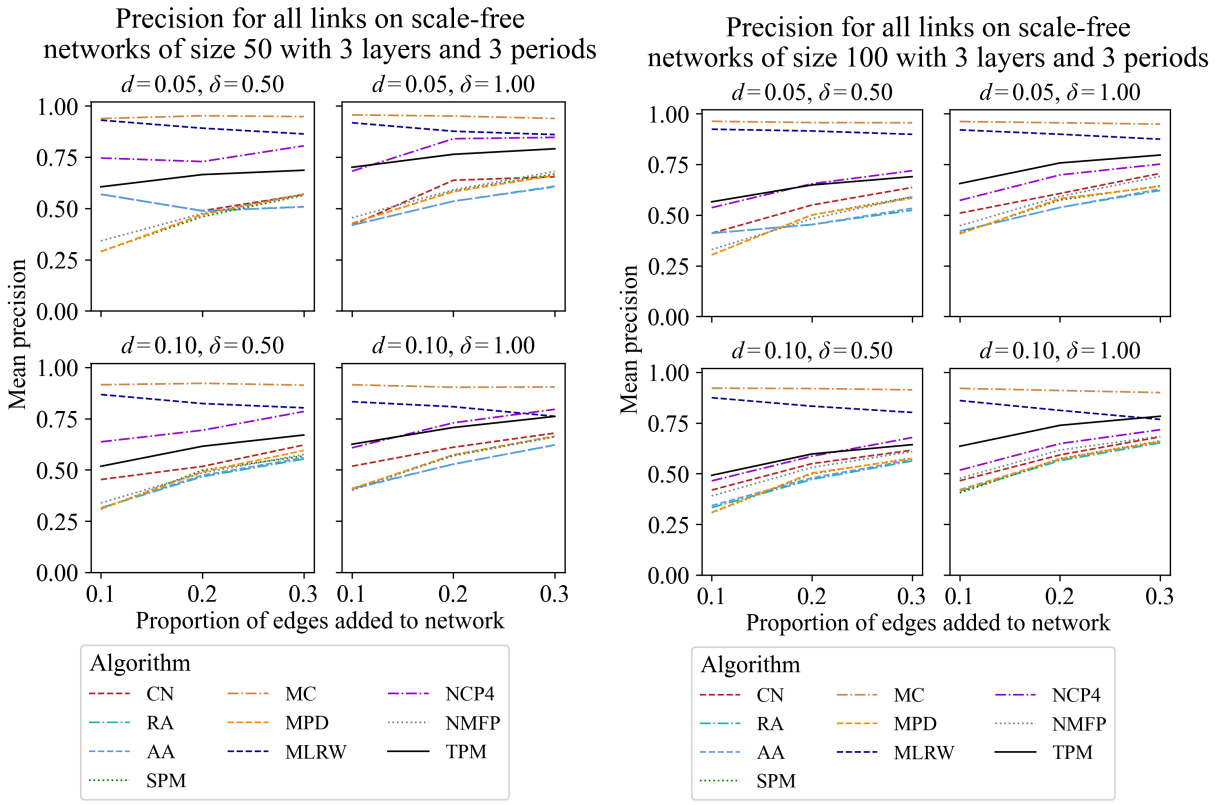
Figure F.16: Runtime for identifying any spurious links in dynamic multiplex small-world networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges.



(a) Fifty-node networks

(b) Hundred-node networks

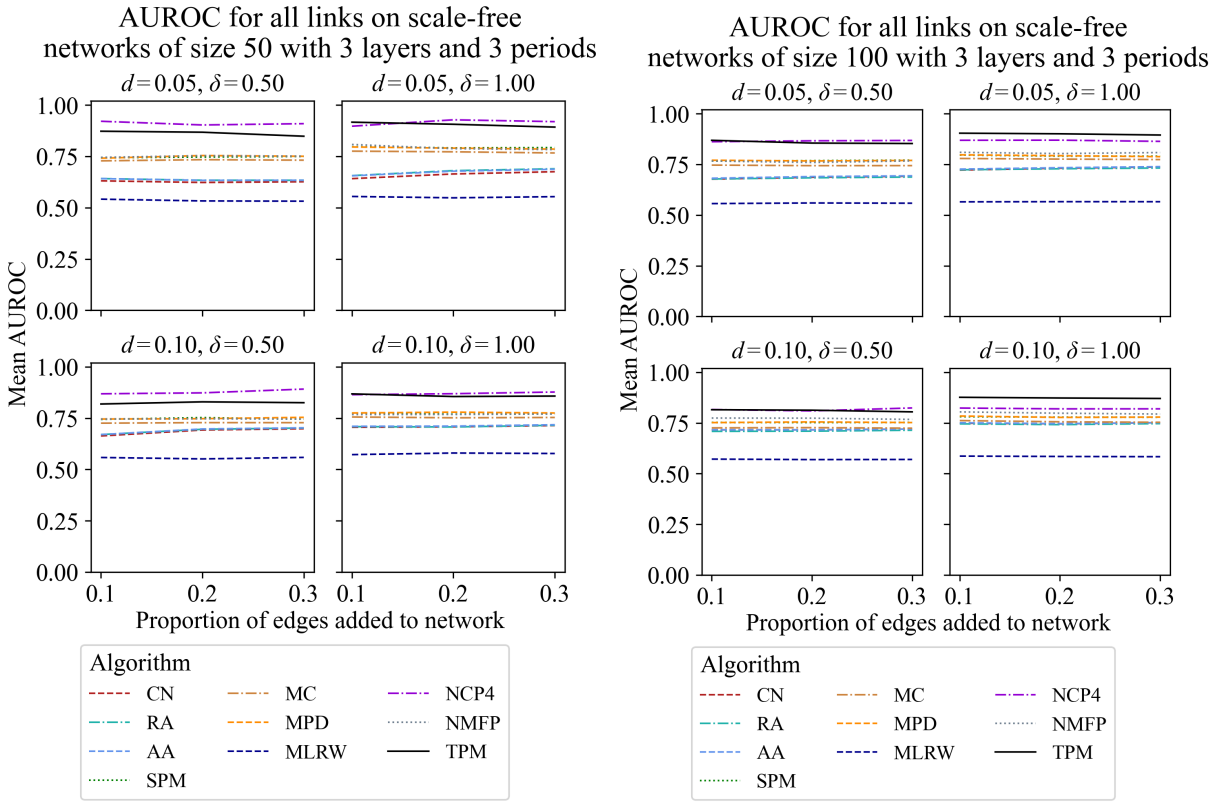
Figure F.17: Area under the precision-recall curve (AUPRC) for identifying any spurious links in dynamic multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

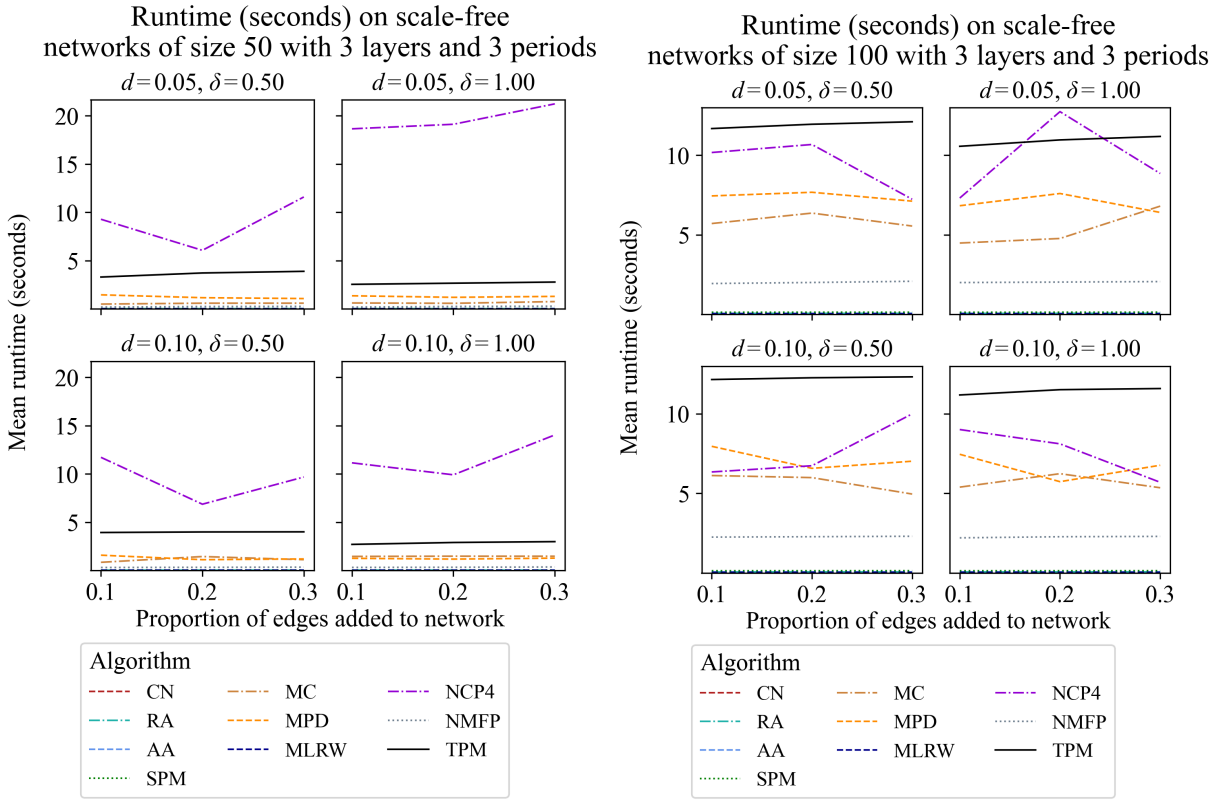
Figure F.18: Precision for identifying any spurious links in dynamic multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

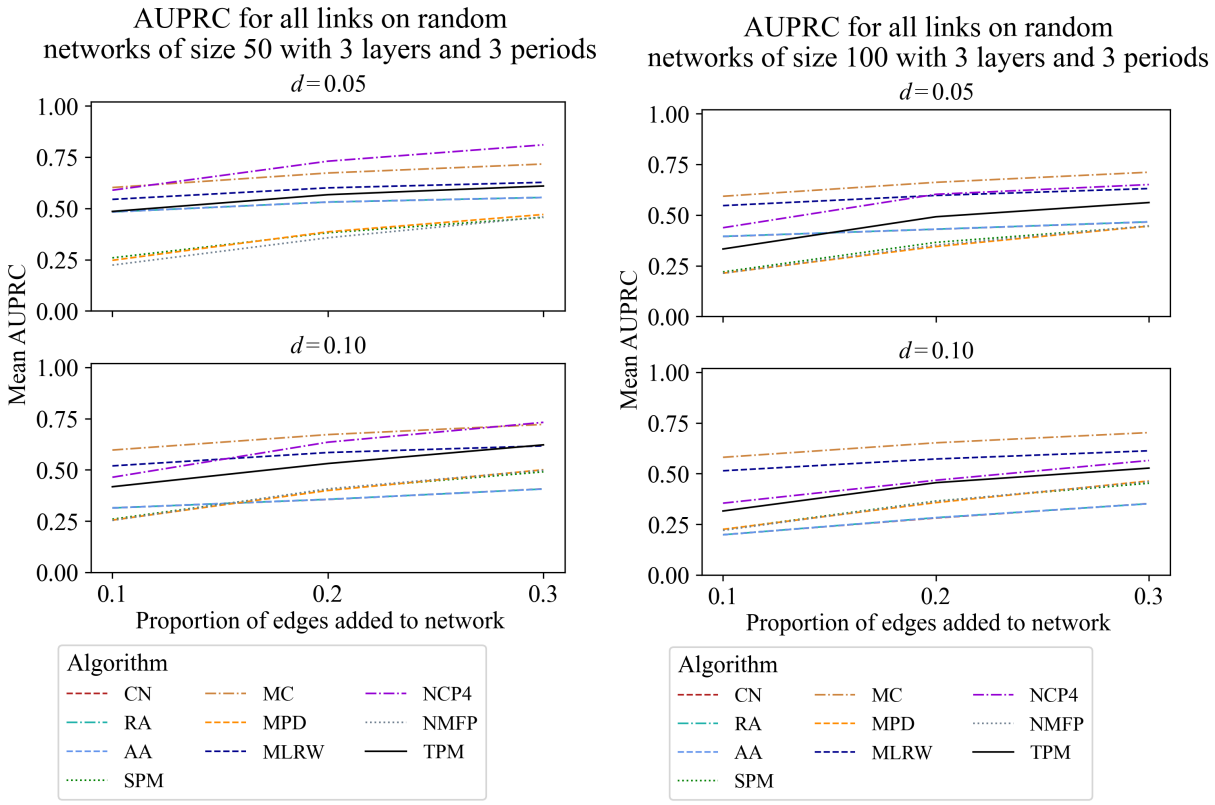
Figure F.19: Area under the receiver operating characteristic (AUROC) for identifying any spurious links in dynamic multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

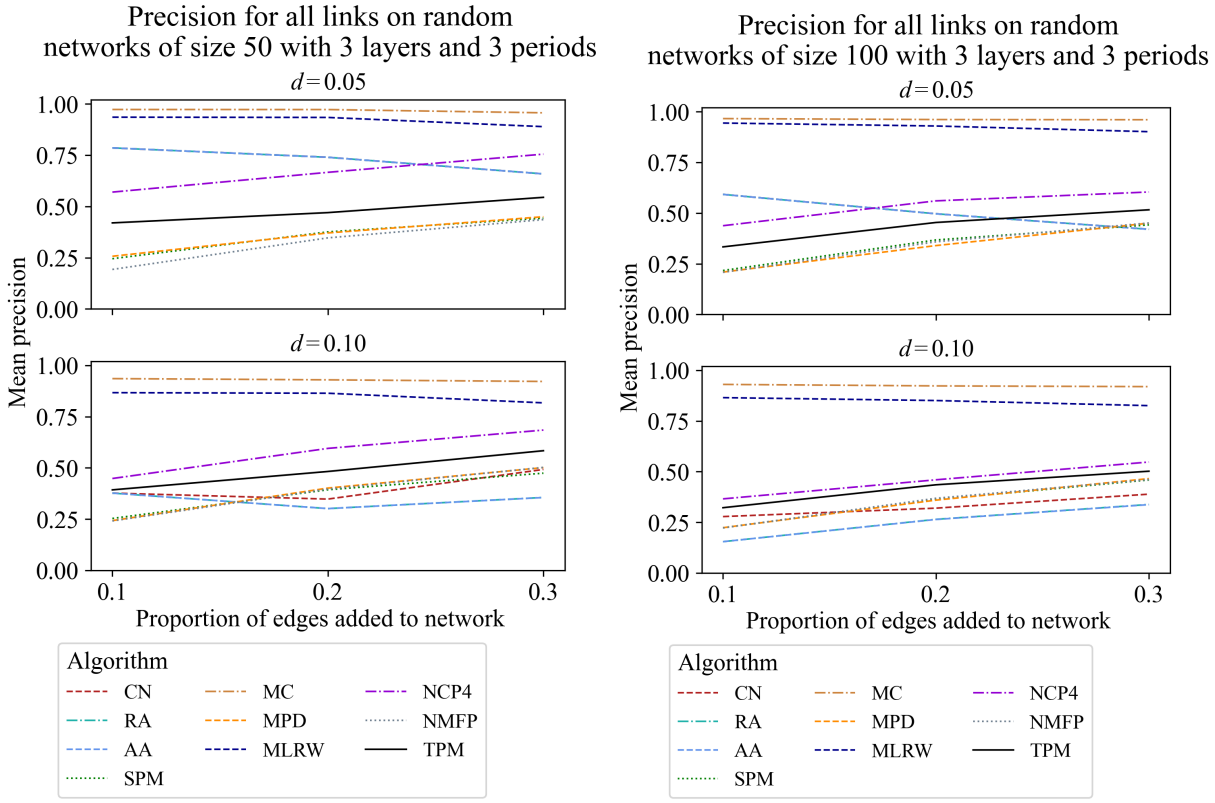
Figure F.20: Runtime for identifying any spurious links in dynamic multiplex scale-free networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights.



(a) Fifty-node networks

(b) Hundred-node networks

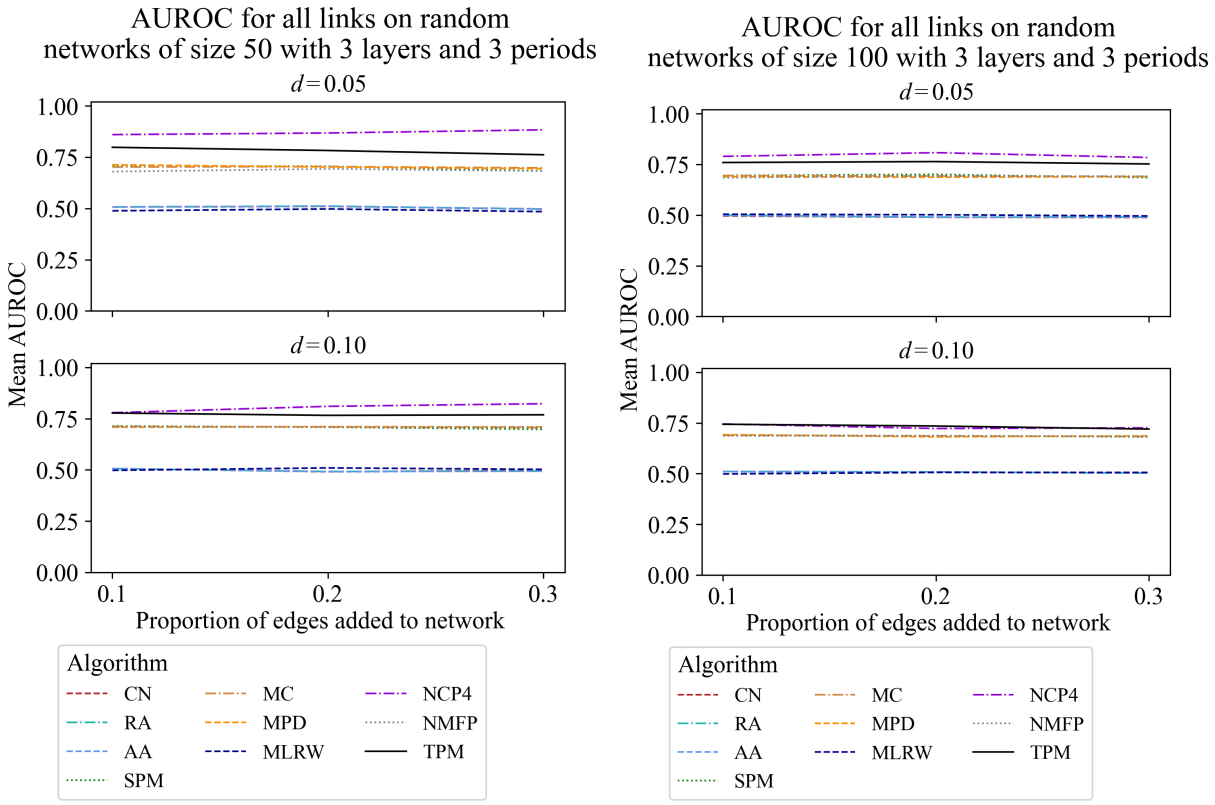
Figure F.21: Area under the precision-recall curve (AUPRC) for identifying any spurious links in dynamic multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

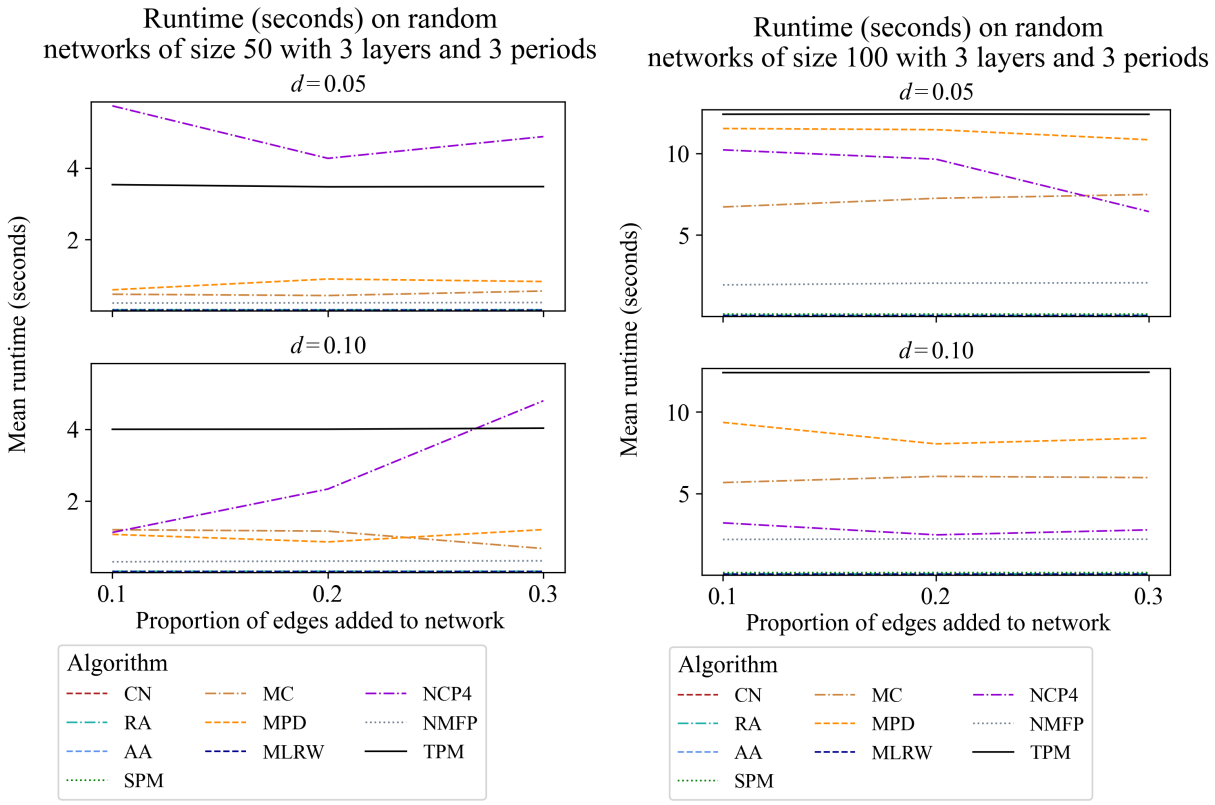
Figure F.22: Precision for identifying any spurious links in dynamic multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

Figure F.23: Area under the receiver operating characteristic (AUROC) for identifying any spurious links in dynamic multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

Figure F.24: Runtime for identifying any spurious links in dynamic multiplex uniformly random networks with three layers. The horizontal axes show the proportion of total edge weight provided as input. Each subplot presents a different network configuration, with d indicating network density.

Appendix G

Confidence Interval Results[1]

G.1 Average Degree Centrality

Figures G.1, G.2, and G.3 show the confidence interval coverages for average degree centrality on small-world, scale-free, and uniformly random networks, respectively.

G.2 Average Betweenness Centrality

Figures G.4, G.5, and G.6 show the confidence interval coverages for average betweenness centrality on small-world, scale-free, and uniformly random networks, respectively.

G.3 Triangle Density

Figures G.7, G.8, and G.9 show the confidence interval coverages for triangle density on small-world, scale-free, and uniformly random networks, respectively.

G.4 Degree Centralities

Figures G.10, G.11, and G.12 show the confidence interval coverages for node-level degree centralities on small-world, scale-free, and uniformly random networks, respectively.

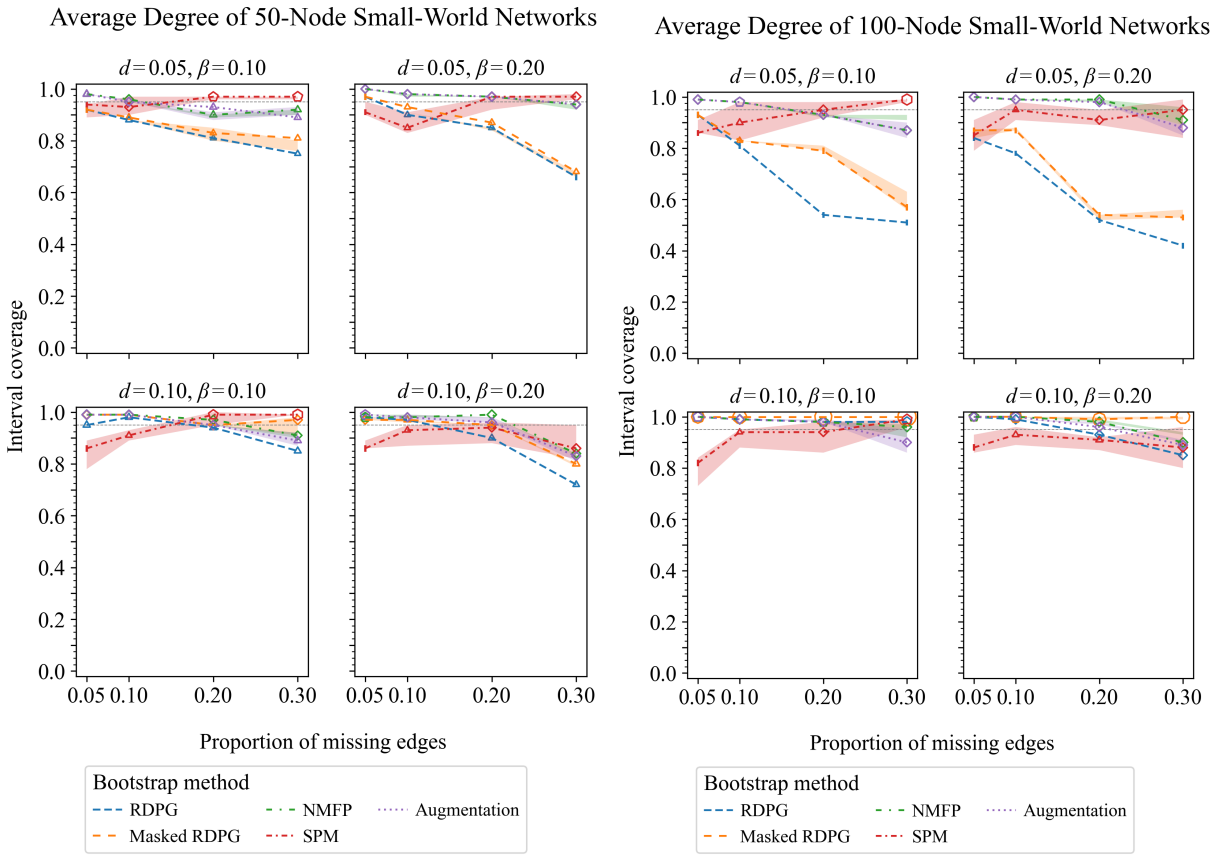
G.5 Betweenness Centralities

Figures G.13, G.14, and G.15 show the confidence interval coverages for node-level betweenness centralities on small-world, scale-free, and uniformly random networks, respectively.

[1]Parts of this appendix were published in *Social, Cultural, and Behavioral Modeling: 18th International Conference, SBP-BRiMS 2025* [16]. The article is available on request.

G.6 Runtime

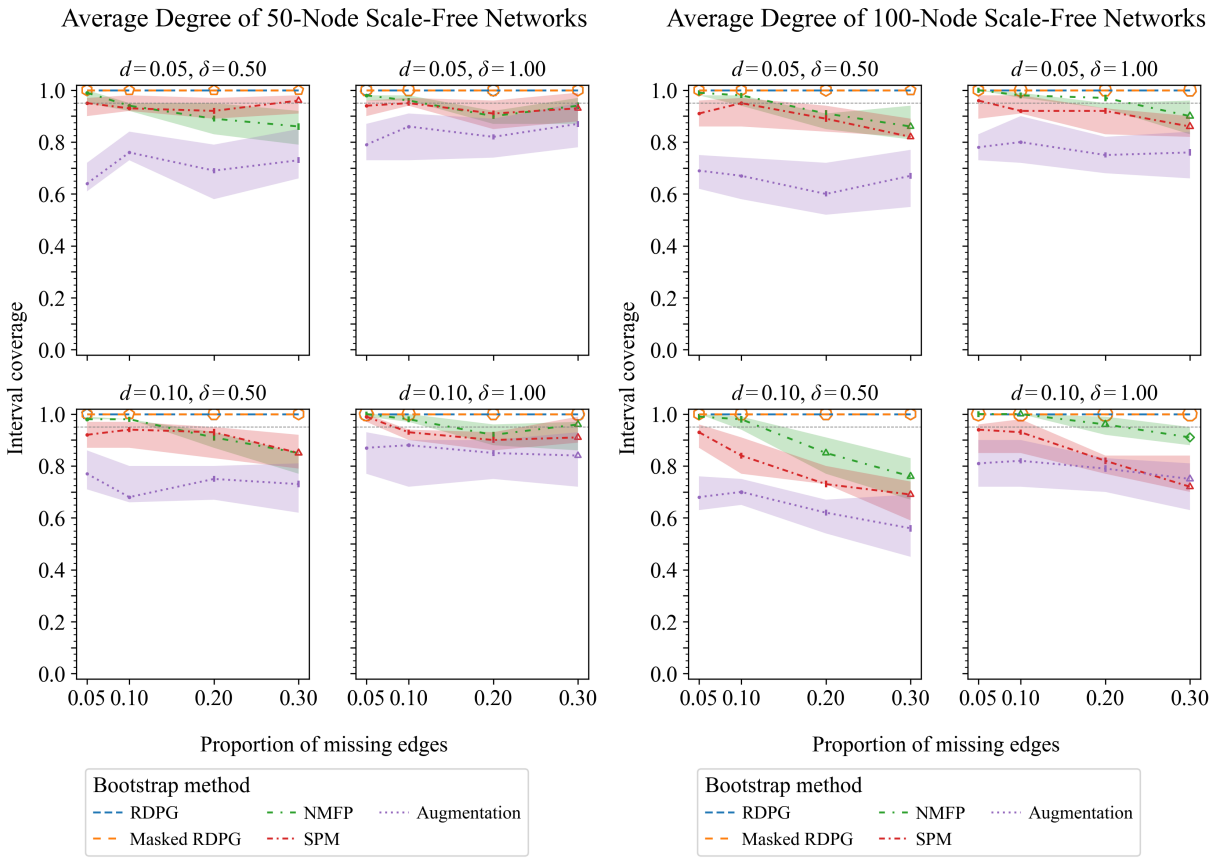
Figures G.16, G.17, and G.18 show the runtimes of the different bootstrapping methods on small-world, scale-free, and uniformly random networks, respectively.



(a) Fifty-node networks

(b) Hundred-node networks

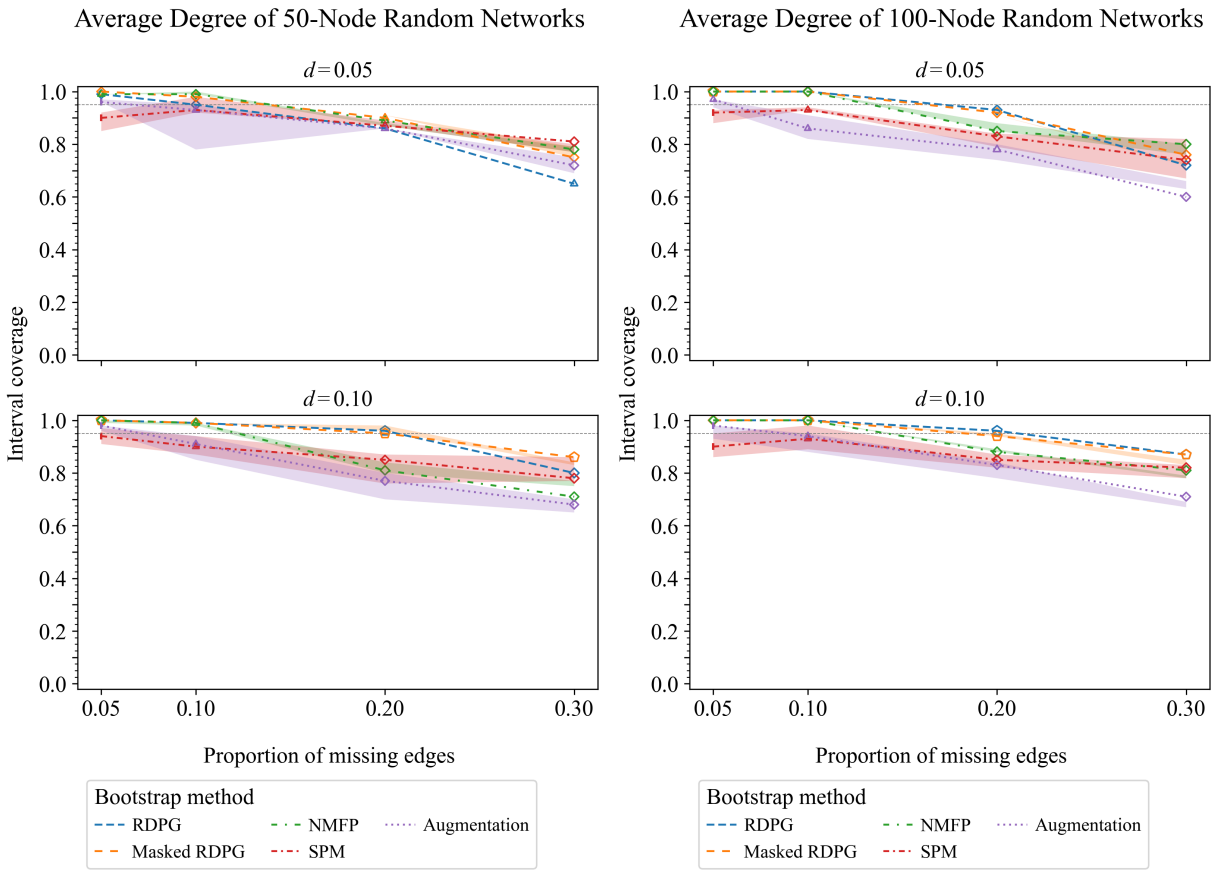
Figure G.1: Confidence interval coverages for average degree centrality on small-world networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges. Augmentation is the strategy of randomly adding edges.



(a) Fifty-node networks

(b) Hundred-node networks

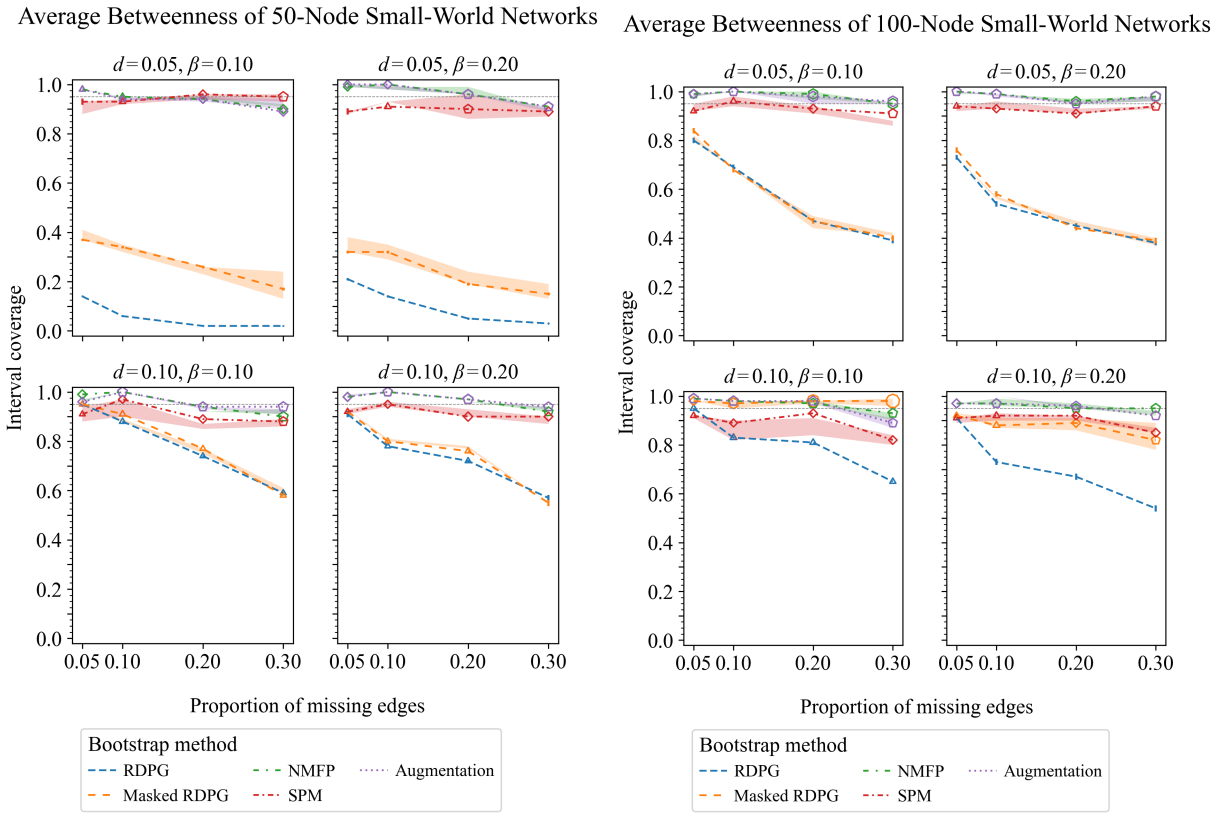
Figure G.2: Confidence interval coverages for average degree centrality on scale-free networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights. Augmentation is the strategy of randomly adding edges.



(a) Fifty-node networks

(b) Hundred-node networks

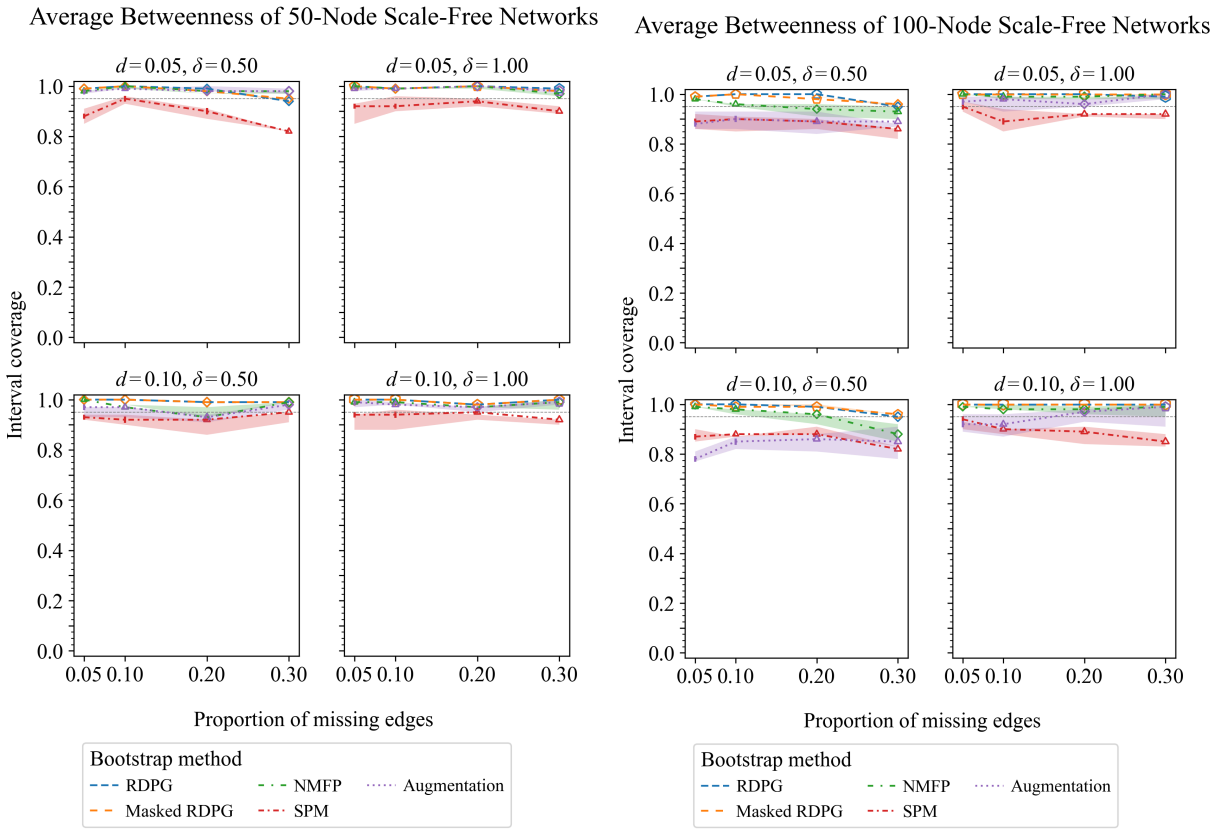
Figure G.3: Confidence interval coverages for average degree centrality on uniformly random networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

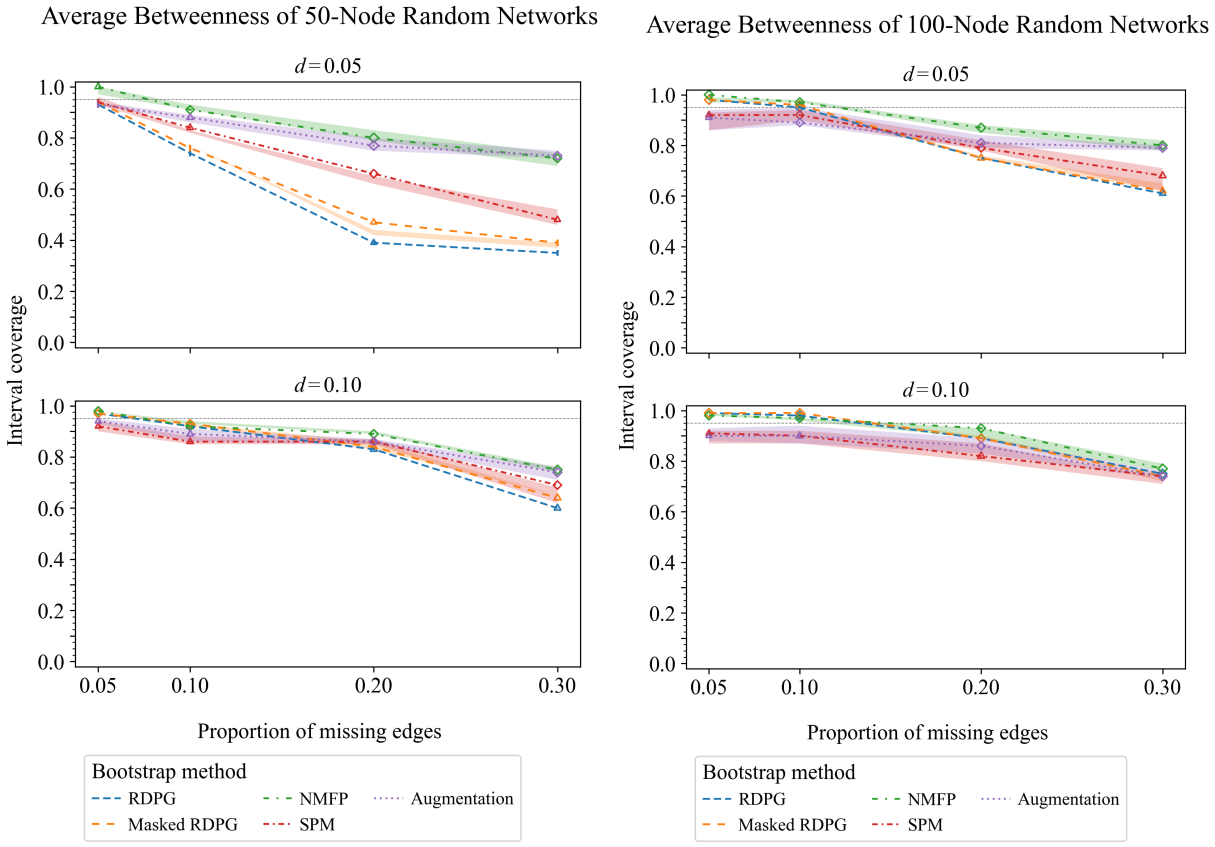
Figure G.4: Confidence interval coverages for average betweenness centrality on small-world networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges. Augmentation is the strategy of randomly adding edges.



(a) Fifty-node networks

(b) Hundred-node networks

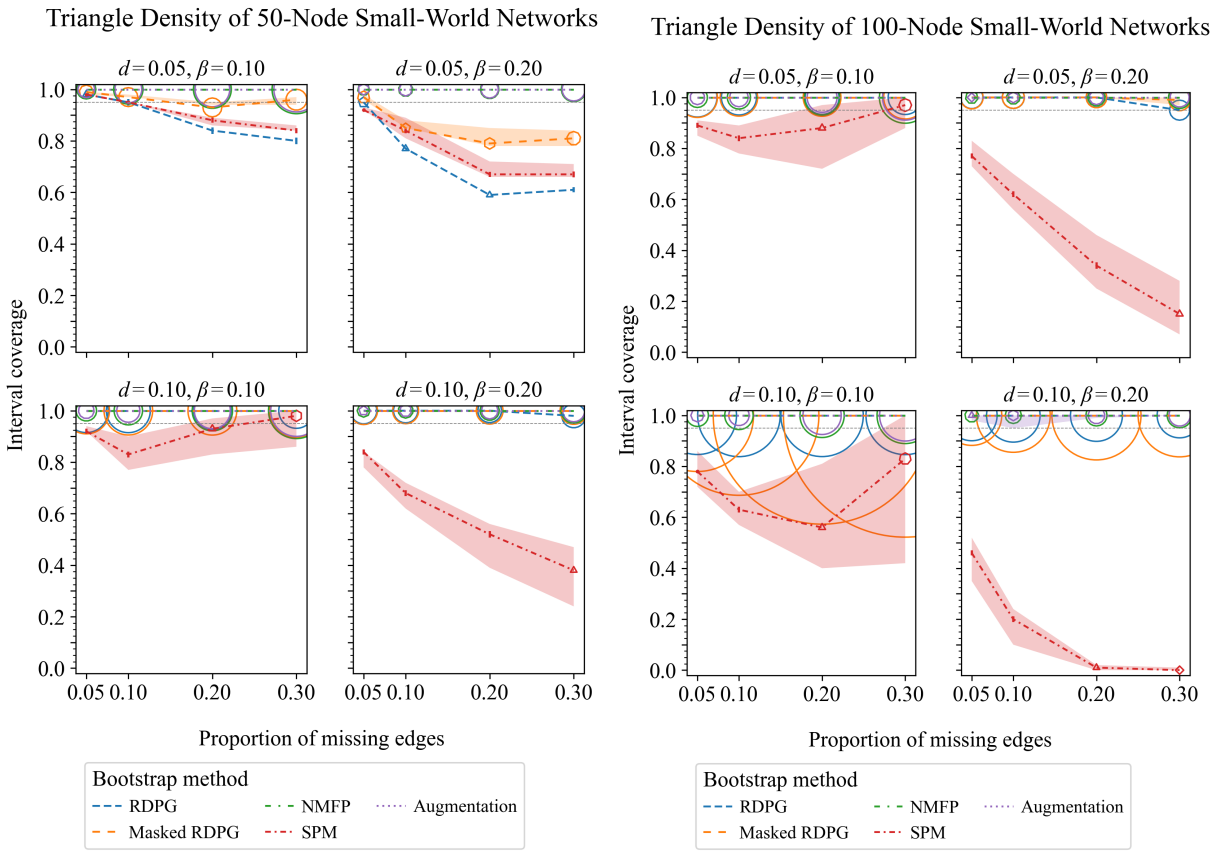
Figure G.5: Confidence interval coverages for average betweenness centrality on scale-free networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights. Augmentation is the strategy of randomly adding edges.



(a) Fifty-node networks

(b) Hundred-node networks

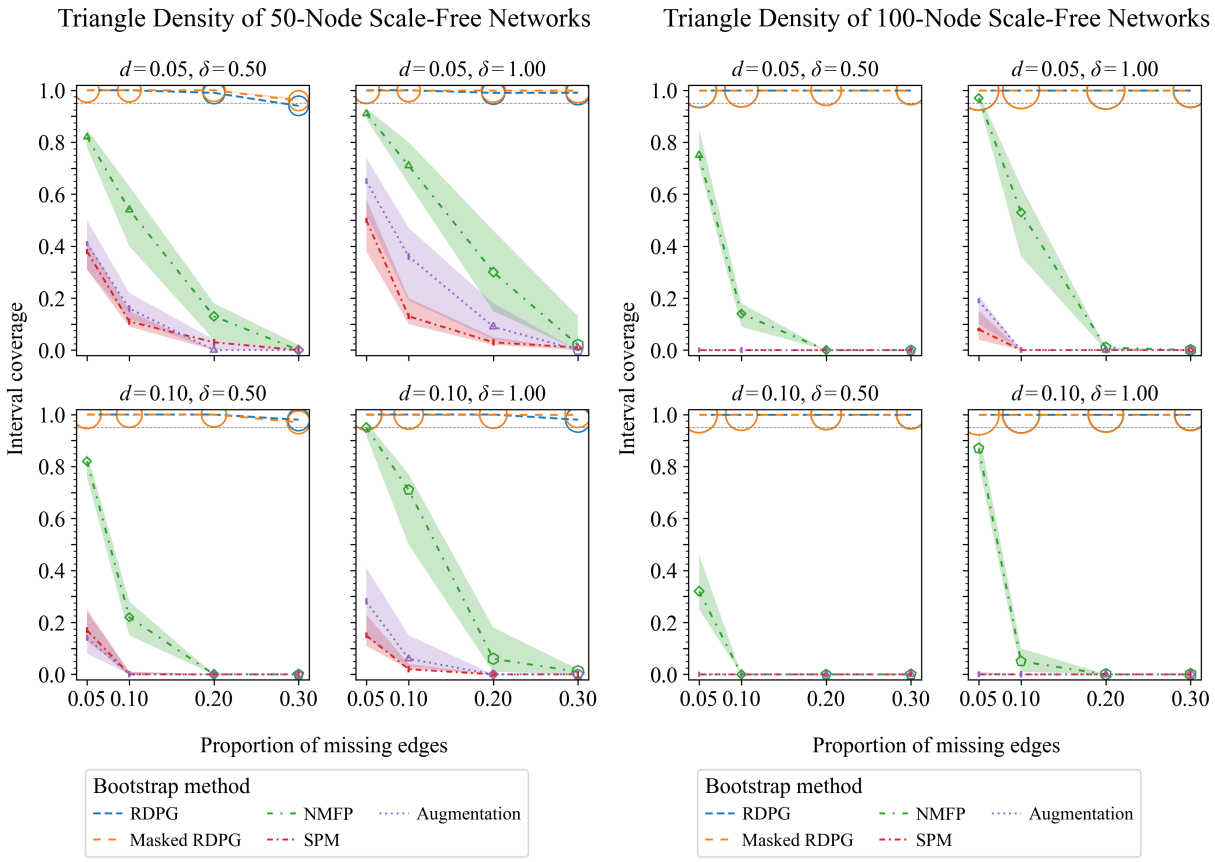
Figure G.6: Confidence interval coverages for average betweenness centrality on uniformly random networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

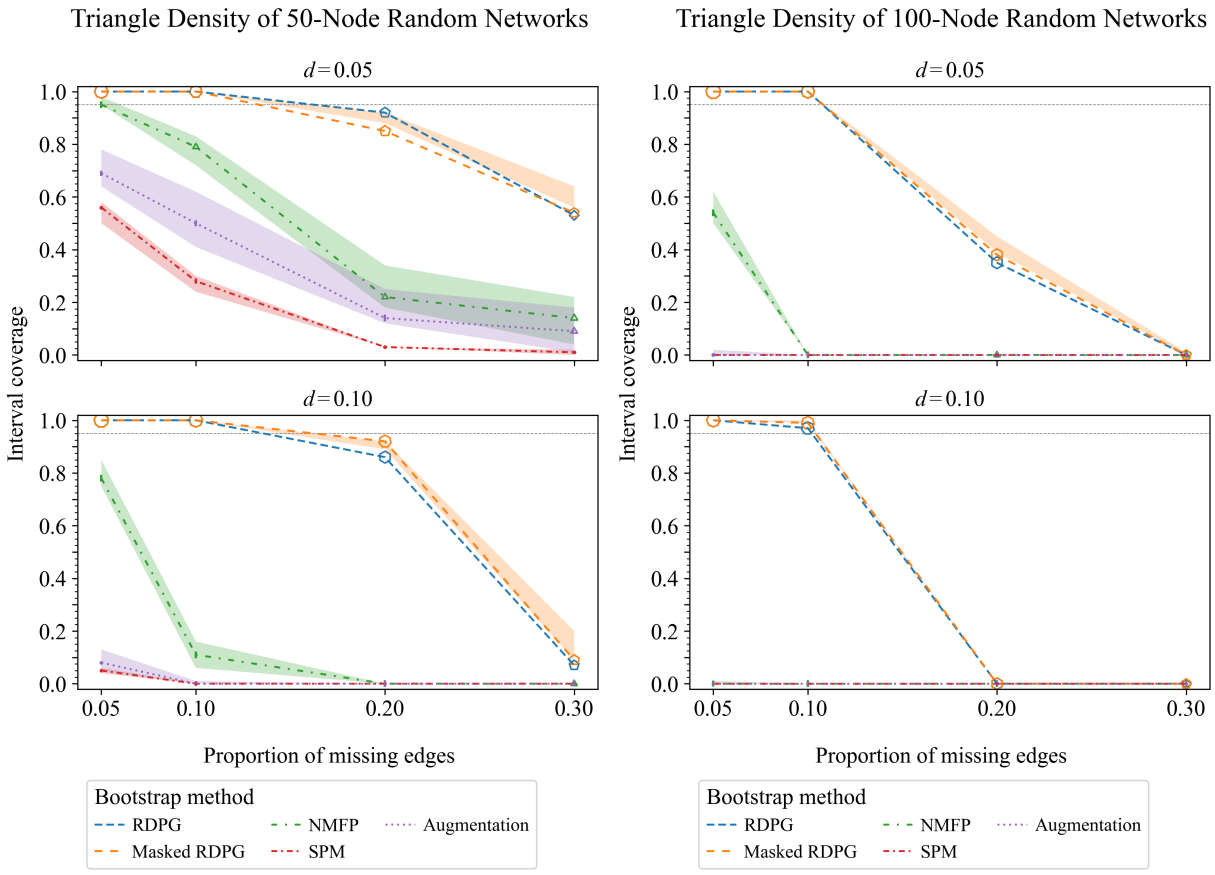
Figure G.7: Confidence interval coverages for triangle density on small-world networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges. Augmentation is the strategy of randomly adding edges.



(a) Fifty-node networks

(b) Hundred-node networks

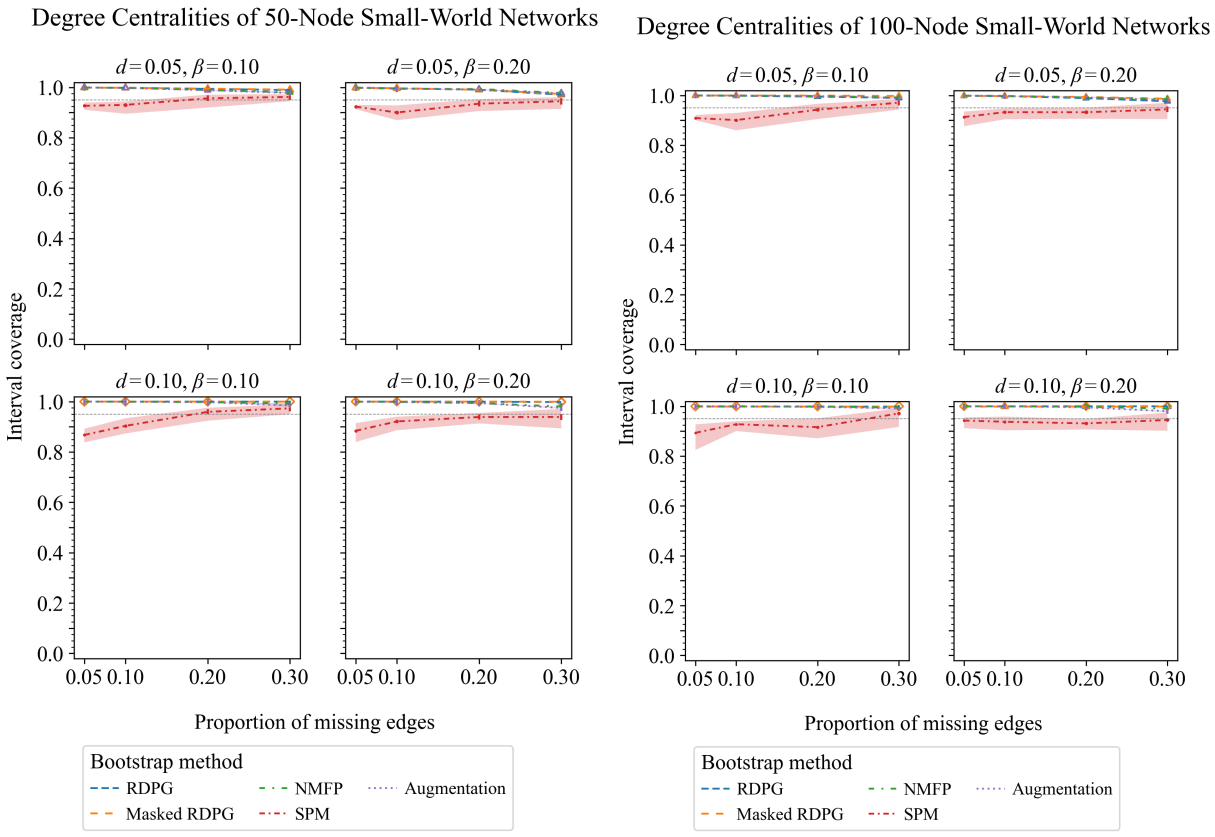
Figure G.8: Confidence interval coverages for triangle density on scale-free networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights. Augmentation is the strategy of randomly adding edges.



(a) Fifty-node networks

(b) Hundred-node networks

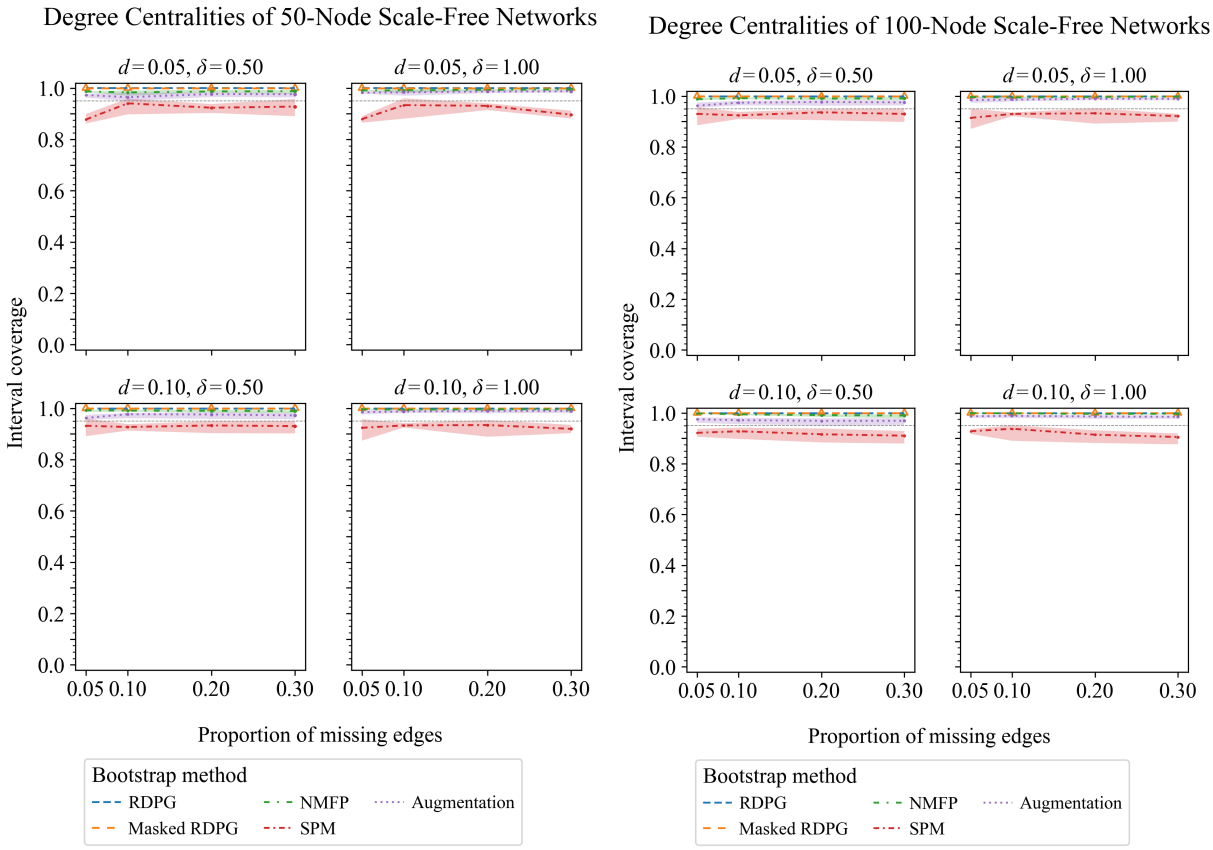
Figure G.9: Confidence interval coverages for triangle density on uniformly random networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

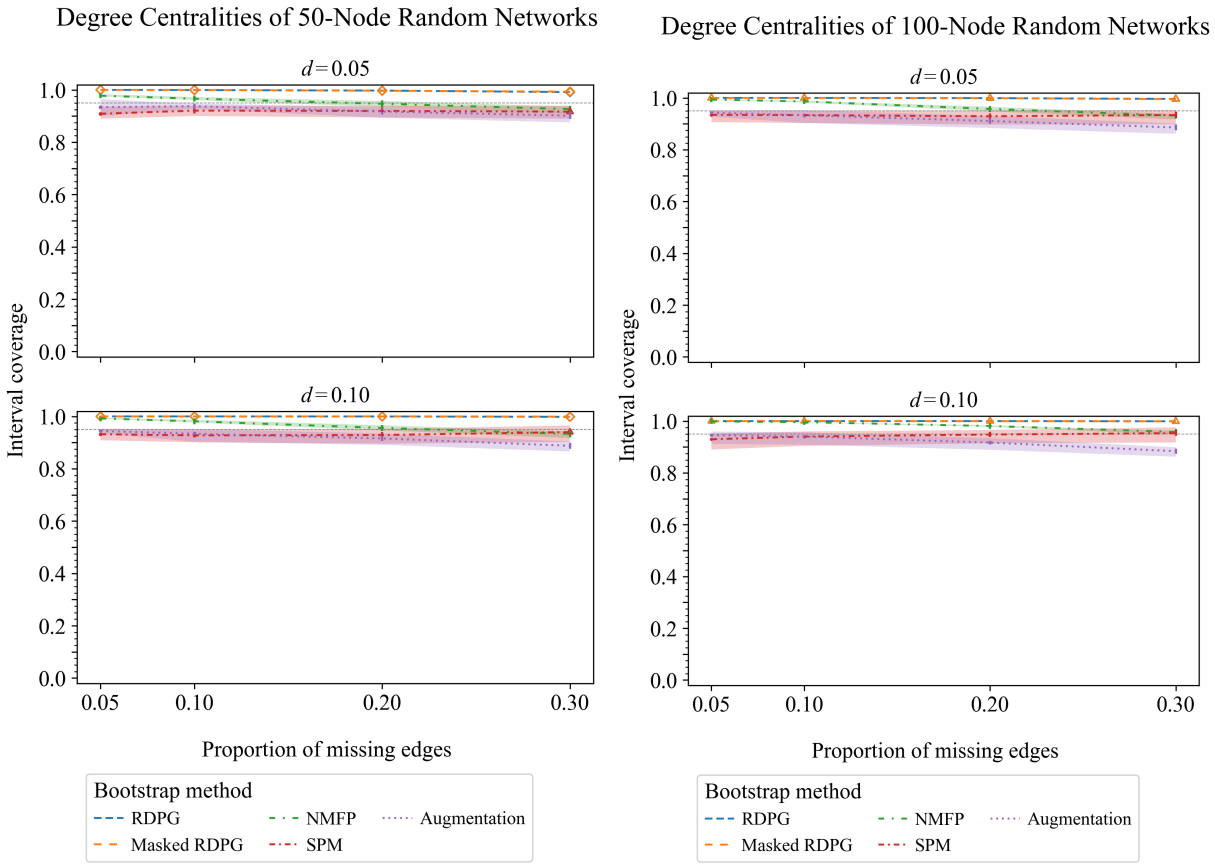
Figure G.10: Confidence interval coverages for node-level degree centralities on small-world networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges. Augmentation is the strategy of randomly adding edges.



(a) Fifty-node networks

(b) Hundred-node networks

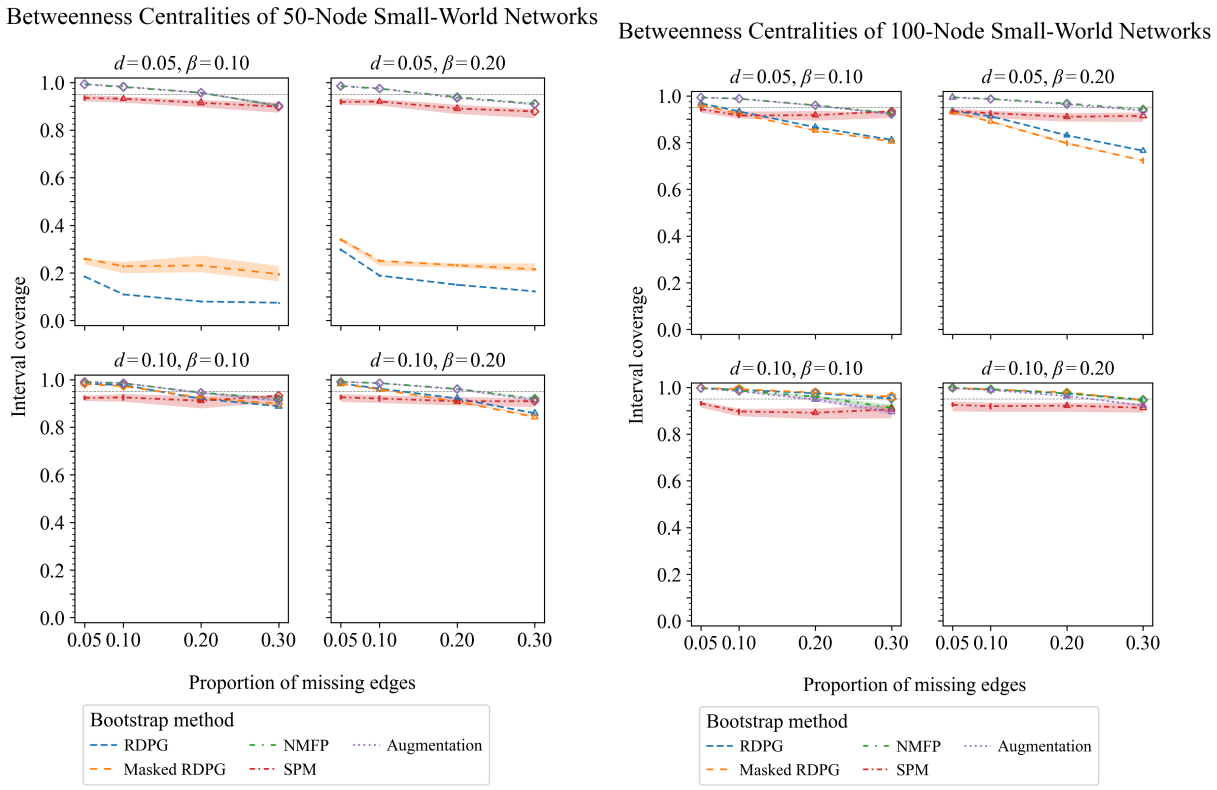
Figure G.11: Confidence interval coverages for node-level degree centralities on scale-free networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights. Augmentation is the strategy of randomly adding edges.



(a) Fifty-node networks

(b) Hundred-node networks

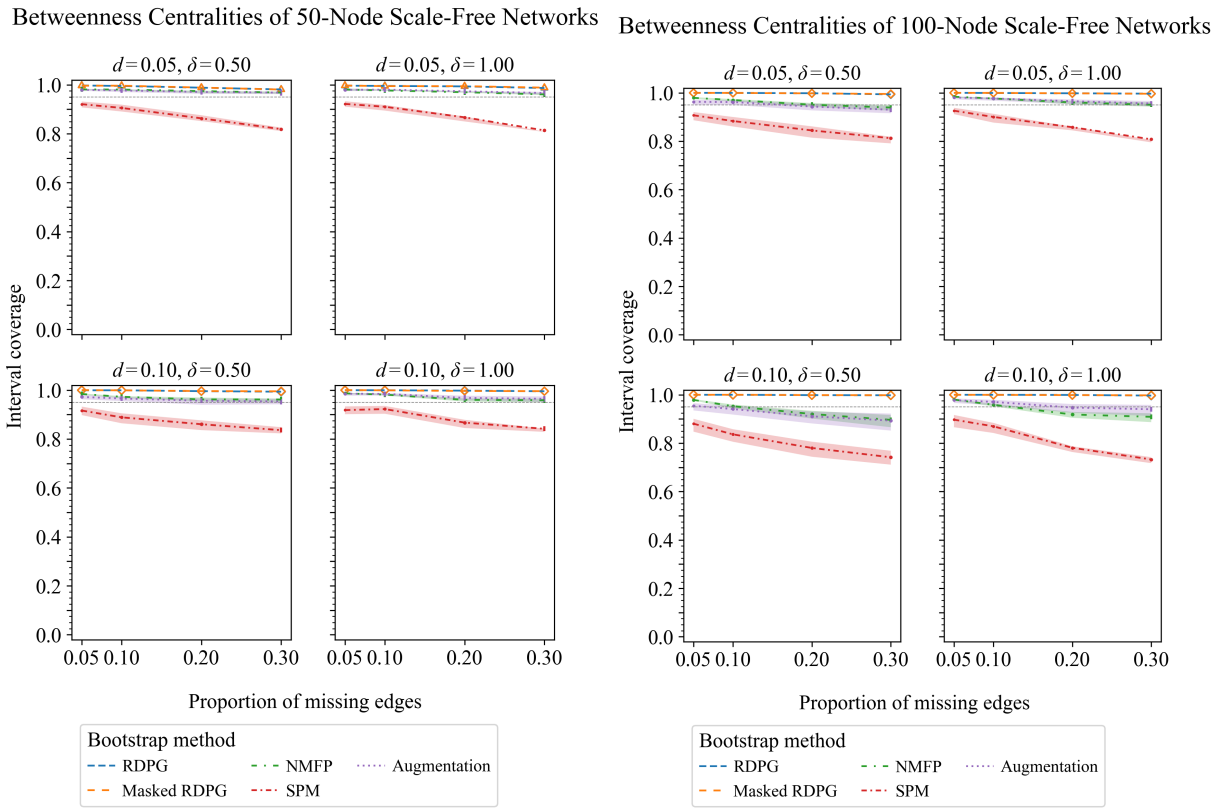
Figure G.12: Confidence interval coverages for node-level degree centralities on uniformly random networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

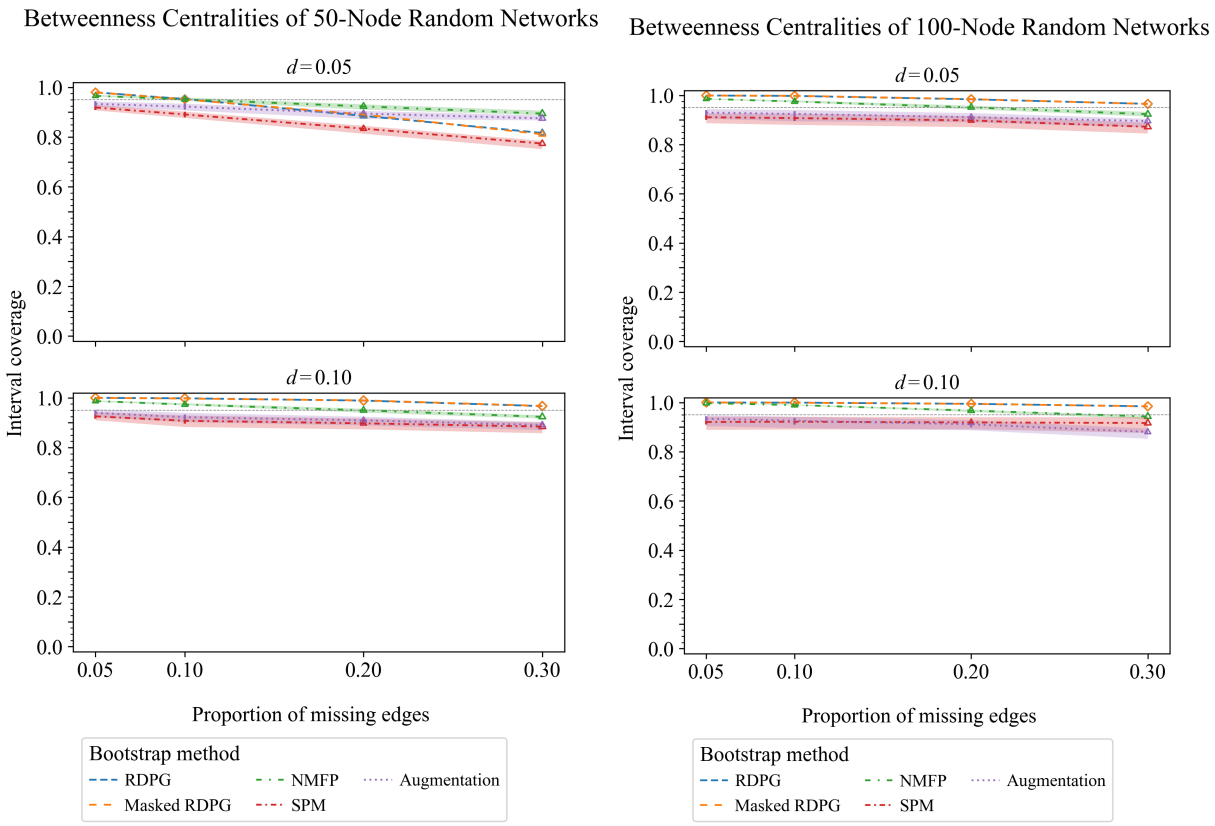
Figure G.13: Confidence interval coverages for node-level betweenness centralities on small-world networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges. Augmentation is the strategy of randomly adding edges.



(a) Fifty-node networks

(b) Hundred-node networks

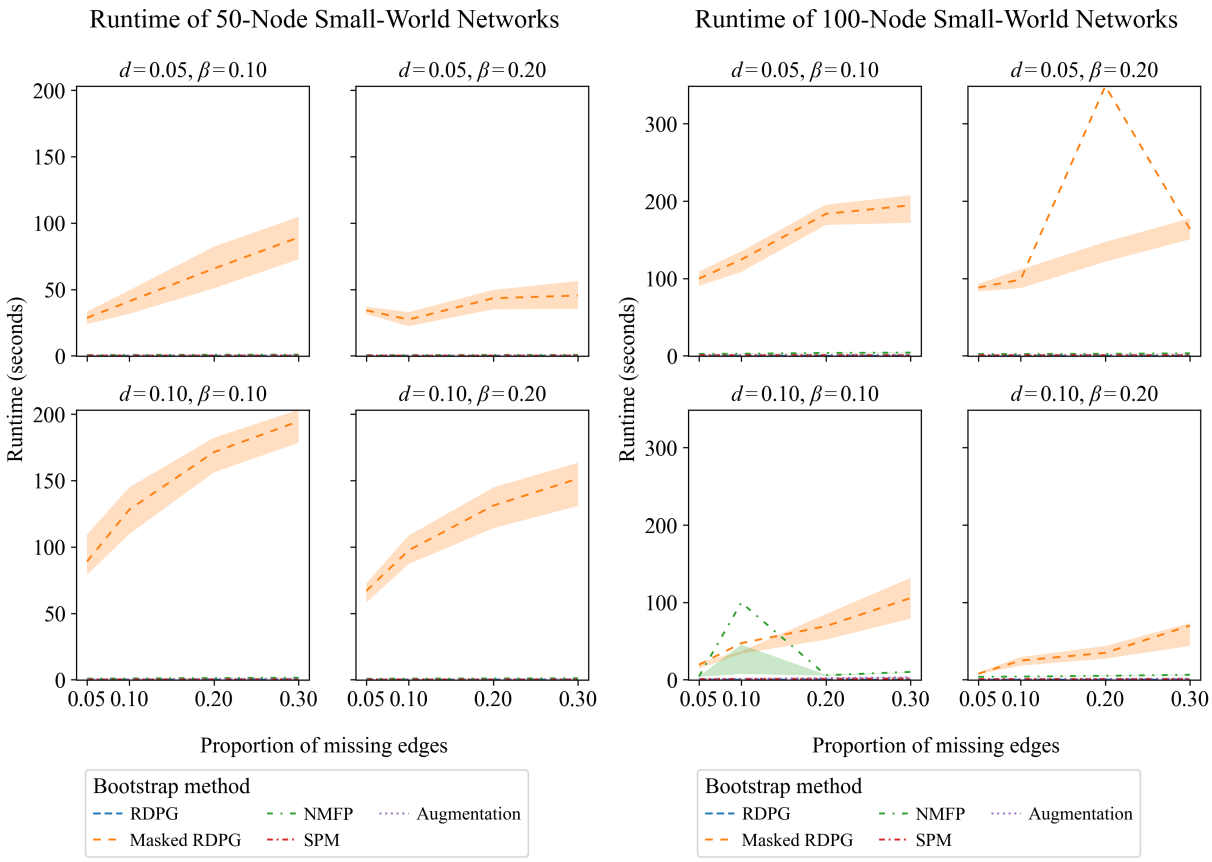
Figure G.14: Confidence interval coverages for node-level betweenness centralities on scale-free networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights. Augmentation is the strategy of randomly adding edges.



(a) Fifty-node networks

(b) Hundred-node networks

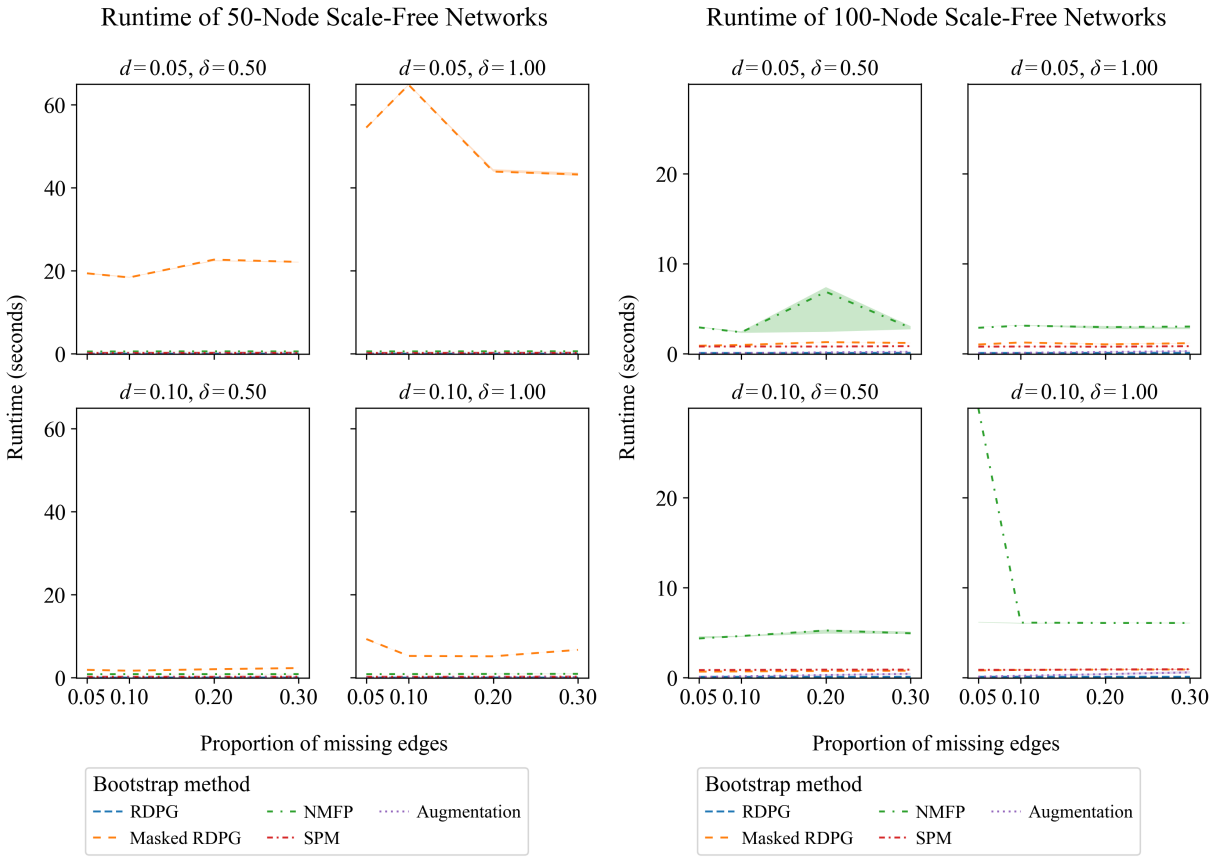
Figure G.15: Confidence interval coverages for node-level betweenness centralities on uniformly random networks. Shaded areas indicate the spread in each algorithm's coverage when the assumption of the amount of missing data is 25% too low or 25% too high. Marker sizes and shapes are proportional to normalized interval widths, with each normalized width being less than the corresponding polygon's number of sides. Dots indicate normalized widths less than one, vertical bars indicate normalized widths less than two, triangles indicate normalized widths less than three, etc. Each subplot presents a different network configuration, with d indicating network density.



(a) Fifty-node networks

(b) Hundred-node networks

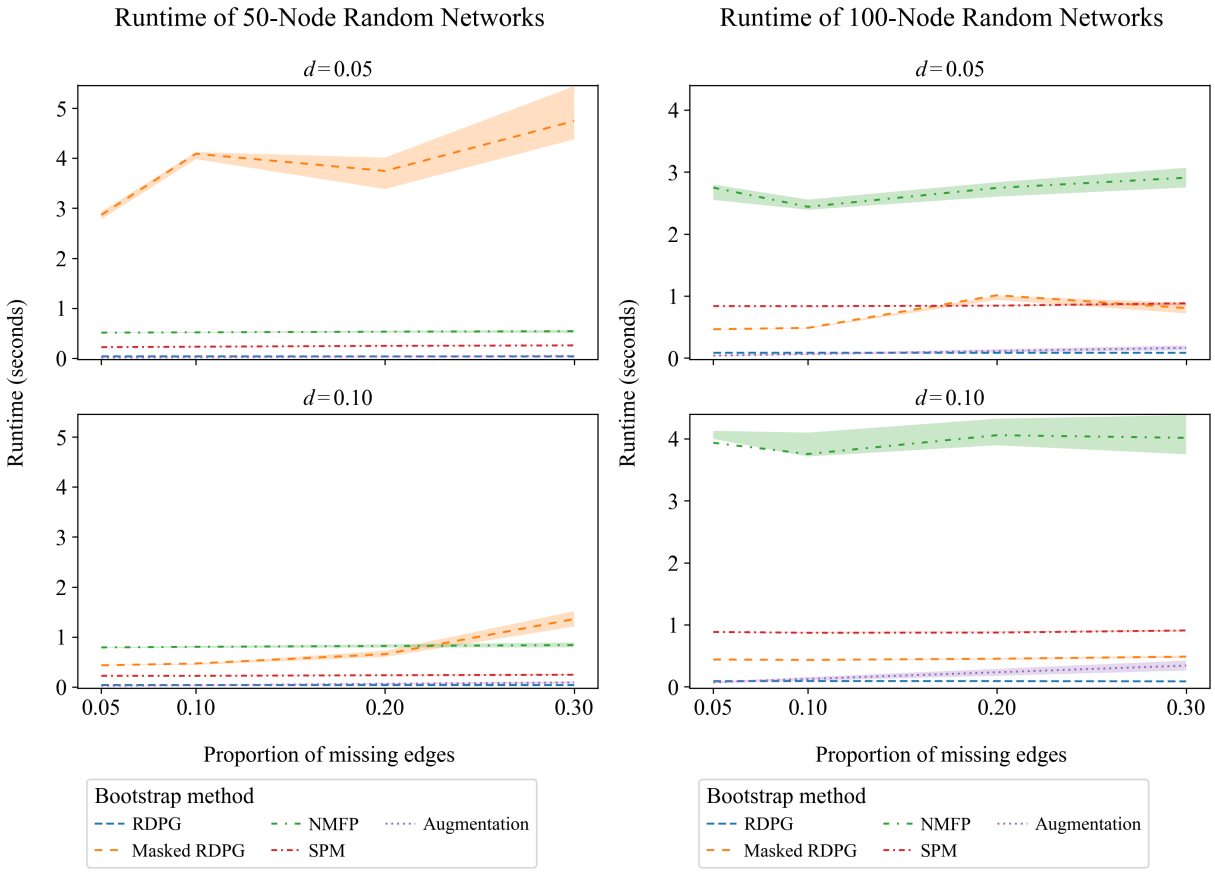
Figure G.16: Runtimes for bootstrapping small-world networks. Shaded areas indicate the spread in each algorithm's runtime when the assumption of the amount of missing data is 25% too low or 25% too high. Each subplot presents a different network configuration, with d indicating network density and β being the probability of "long-distance" edges. Augmentation is the strategy of randomly adding edges.



(a) Fifty-node networks

(b) Hundred-node networks

Figure G.17: Runtimes for bootstrapping scale-free networks. Shaded areas indicate the spread in each algorithm's runtime when the assumption of the amount of missing data is 25% too low or 25% too high. Each subplot presents a different network configuration, with d indicating network density and δ dictating the accumulation of edge weights. Augmentation is the strategy of randomly adding edges.



(a) Fifty-node networks

(b) Hundred-node networks

Figure G.18: Runtimes for bootstrapping uniformly random networks. Shaded areas indicate the spread in each algorithm's runtime when the assumption of the amount of missing data is 25% too low or 25% too high. Each subplot presents a different network configuration, with d indicating network density. Augmentation is the strategy of randomly adding edges.

Appendix H

Regression Results for Account Creation Bursts in 2020 French Terror Tweets

H.1 Bot Activity

Tables H.1 through H.3 contain the complete specifications for the regression models fit to predict user bot probabilities, whereas Tables H.4 through H.6 contain information on the corresponding average marginal effects.

H.2 Sharing of Low-Credibility Sources

Tables H.7 through H.9 contain the complete specifications for the regression models fit to predict each user's proportion of tweets containing a link to a low-credibility site, whereas Tables H.10 through H.12 contain information on the corresponding average marginal effects.

Bot probability, PFA = 0.25

Variable	Coefficient	SE	z	p	Corrected p
Intercept	0.806884	0.021067	38.300765	0e+00	
Burst	−0.139539	0.073793	−1.890968	0.058629	0.127591
N	1000				
Fit RMSD	0.136797				
$\frac{\text{RMSD}}{\sigma}$	0.997631				

Table H.1: Specification for a one-part fractional regression model fit to predict each user's bot probability. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Bot probability, PFA = 0.30

Variable	Coefficient	SE	z	p	Corrected p
Intercept	0.827412	0.022356	37.010656	8e−300	
Burst	−0.159869	0.051295	−3.116666	0.001829	0.013115
N	1000				
Fit RMSD	0.136351				
$\frac{\text{RMSD}}{\sigma}$	0.994374				

Table H.2: Specification for a one-part fractional regression model fit to predict each user's bot probability. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Bot probability, PFA = 0.35

Variable	Coefficient	SE	z	p	Corrected p
Intercept	0.811804	0.026553	30.572770	3e−205	
Burst	−0.037905	0.040914	−0.926457	0.354209	0.532118
N	1000				
Fit RMSD	0.136994				
$\frac{\text{RMSD}}{\sigma}$	0.999067				

Table H.3: Specification for a one-part fractional regression model fit to predict each user's bot probability. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Bot probability, PFA = 0.25

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	−0.030537	0.016474	−1.853606	0.063796	0.127591	−0.222698
N	1000					
Fit RMSD	0.136797					
$\frac{\text{RMSD}}{\sigma}$	0.997631					

Table H.4: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict each user's bot probability. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Bot probability, PFA = 0.30

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	-0.034855	0.011376	-3.063747	0.002186	0.013115	-0.254187
N	1000					
Fit RMSD	0.136351					
$\frac{\text{RMSD}}{\sigma}$	0.994374					

Table H.5: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict each user's bot probability. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Bot probability, PFA = 0.35

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	-0.008130	0.008785	-0.925424	0.354745	0.532118	-0.059290
N	1000					
Fit RMSD	0.136994					
$\frac{\text{RMSD}}{\sigma}$	0.999067					

Table H.6: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict each user's bot probability. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of posts with low-credibility links, PFA = 0.25

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-7.496983	0.171004	-43.840970	$0e{+}00$	
Burst	-0.819715	0.422968	-1.938008	0.052622	0.127591
N	1000				
Fit RMSD	0.002762				
$\frac{\text{RMSD}}{\sigma}$	0.999026				

Table H.7: Specification for a one-part fractional regression model fit to predict each user's proportion of posts with low-credibility links. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of posts with low-credibility links, PFA = 0.30

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-7.619400	0.162136	-46.994009	$0e{+}00$	
Burst	0.334927	0.470414	0.711983	0.476475	0.635301
N	1000				
Fit RMSD	0.002762				
$\frac{\text{RMSD}}{\sigma}$	0.999106				

Table H.8: Specification for a one-part fractional regression model fit to predict each user's proportion of posts with low-credibility links. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of posts with low-credibility links, PFA = 0.35

Variable	Coefficient	SE	z	p	Corrected p
Intercept	-7.608861	0.202372	-37.598428	$0e{+}00$	
Burst	0.142761	0.335000	0.426153	0.669997	0.677069
N	1000				
Fit RMSD	0.002763				
$\frac{\text{RMSD}}{\sigma}$	0.999407				

Table H.9: Specification for a one-part fractional regression model fit to predict each user's proportion of posts with low-credibility links. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable coefficient is shown with its robust standard error (SE), z-score, raw p-value, and corrected p-value. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of posts with low-credibility links, PFA = 0.25

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	-0.000310	0.000134	-2.317179	0.020494	0.081976	-0.112177
N	1000					
Fit RMSD	0.002762					
$\frac{\text{RMSD}}{\sigma}$	0.999026					

Table H.10: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict each user's proportion of posts with low-credibility links. Account creation bursts were detected with a probability of false alarm (PFA) of 0.25. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of posts with low-credibility links, PFA = 0.30

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.000195	0.000313	0.623472	0.532975	0.639569	0.070554
N	1000					
Fit RMSD	0.002762					
$\frac{\text{RMSD}}{\sigma}$	0.999106					

Table H.11: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict each user's proportion of posts with low-credibility links. Account creation bursts were detected with a probability of false alarm (PFA) of 0.30. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Proportion of posts with low-credibility links, PFA = 0.35

Variable	AME	SE	z	p	Corrected p	$\frac{\text{AME}}{\sigma}$
Burst	0.000076	0.000183	0.416466	0.677069	0.677069	0.027506
N	1000					
Fit RMSD	0.002763					
$\frac{\text{RMSD}}{\sigma}$	0.999407					

Table H.12: Average marginal effects (AMEs) for a one-part fractional regression model fit to predict each user's proportion of posts with low-credibility links. Account creation bursts were detected with a probability of false alarm (PFA) of 0.35. Each variable's AME is shown with its robust standard error (SE), z-score, raw p-value, corrected p-value, and AME divided by the standard deviation of the dependent variable $\left(\frac{\text{AME}}{\sigma}\right)$. The number of data points (N), root-mean-square deviation (RMSD) on the fitted data, and RMSD divided by the standard deviation of the dependent variable $\left(\frac{\text{RMSD}}{\sigma}\right)$ are also shown.

Bibliography

[1] Lada A. Adamic and Eytan Adar. Friends and neighbors on the Web. *Social Networks*, 25 (3):211–230, 2003. doi: 10.1016/S0378-8733(03)00009-1. 3.2.2

[2] Charu C. Aggarwal, Yan Xie, and Philip S. Yu. A framework for dynamic link prediction in heterogeneous networks. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 7:14–33, 2014. doi: 10.1002/sam.11198. 3.1.1, 6.2.3

[3] Ino Agrafioti, Jonathan Swire, James Abbott, Derek Huntley, Sarah Butcher, and Michael P. H. Stumpf. Comparative analysis of the *Saccharomyces cerevisiae* and *Caenorhabditis elegans* protein interaction networks. *BMC Evolutionary Biology*, 5(1):23, 2005. doi: 10.1186/1471-2148-5-23. 4.1.1, 6.1.2

[4] Samer Al-khateeb and Nitin Agarwal. *Deviance in Social Media and Social Cyber Forensics: Uncovering Hidden Relations Using Open Source Information (OSINF)*. SpringerBriefs in Cybersecurity. Springer, 2019. ISBN 9783030136895. doi: 10.1007/978-3-030-13690-1. 1.5, 6.2.2

[5] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74:47–97, 2002. doi: 10.1103/RevModPhys.74.47. 3.2.4

[6] Zack W. Almquist and Carter T. Butts. Dynamic network analysis with missing data: Theory and methods. *Statistica Sinica*, 28(3):1245–1264, 2018. doi: 10.5705/ss.202016.0108. 1.5.1

[7] Sofía Aparicio, Javier Villazón-Terrazas, and Gonzalo Álvarez. A model for scale-free networks: Application to Twitter. *Entropy*, 17(8):5848–5867, 2015. doi: 10.3390/e17085848. 4.5.2, 5.5.2

[8] Humaira Arshad, Aman Jantan, and Esther Omolara. Evidence collection and forensics on social networks: Research challenges and directions. *Digital Investigation*, 28:126–138, 2019. doi: 10.1016/j.diin.2019.02.001. 6.2.2

[9] DFRWS attendees. A road map for digital forensic research. Technical report, First Digital Forensic Research Workshop (DFRWS), 2001. DTR - T001-01 FINAL. 1.5, 6.2.2

[10] Christopher A. Bail, Brian Guay, Emily Maloney, Aiden Combs, D. Sunshine Hillygus, Friedolin Merhout, Deen Freelon, and Alexander Volfovsky. Assessing the Russian Internet Research Agency's impact on the political attitudes and behaviors of American Twitter users in late 2017. *Proceedings of the National Academy of Sciences*, 117(1):243–250, 2019. doi: 10.1073/pnas.1906420116. 1.1, 2.1.1

[11] Michael Barber and John B. Holbein. The participatory and partisan impacts of mandatory vote-by-mail. *Science Advances*, 6(35):eabc7685, 2020. doi: 10.1126/sciadv.abc7685. 2.2.5

[12] Alain Barrat, Marc Barthélemy, and Alessandro Vespignani. Weighted evolving networks: Coupling topology and weight dynamics. *Physical Review Letters*, 92(22):228701, 2004. doi: 10.1103/PhysRevLett.92.228701. 3.2.4

[13] Daniele Bellutta and Kathleen M. Carley. Investigating coordinated account creation using burst detection and network analysis. *Journal of Big Data*, 10:20, 2023. doi: 10.1186/s40537-023-00695-7. **??**, 1, 2.5.2, 1, 1

[14] Daniele Bellutta and Kathleen M. Carley. Improving the reliability of network analysis using link prediction. 2023 Informed Democracy and Social-cybersecurity Conference. [Working Paper], 2023. 1

[15] Daniele Bellutta and Kathleen M. Carley. Improving the prediction of missing links in covert networks. 2024 Informed Democracy and Social-cybersecurity Conference. [Abstract], 2024. 1

[16] Daniele Bellutta and Kathleen M. Carley. Toward evaluating network confidence intervals. In *Social, Cultural, and Behavioral Modeling (SBP-BRiMS 2025)*, pages 183–192, 2025. doi: 10.1007/978-3-032-07715-8_18. **??**, **??**, 1, 1

[17] Daniele Bellutta and Kathleen M. Carley. Improving the prediction of missing links in multiplex networks. 2025 Informed Democracy and Social-cybersecurity Conference. [Working Paper], 2025. 1

[18] Daniele Bellutta, Youdinghuan Chen, Daveed Gartenstein-Ross, Chiara Pulice, Anja Subasic, and V. S. Subrahmanian. Understanding shifting triadic relationships in the Al-Qaeda/ISIS faction ecosystem. *IEEE Transactions on Computational Social Systems*, 7 (6):1423–1434, 2020. doi: 10.1109/TCSS.2020.3022586. 1.5.2

[19] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):289–300, 1995. URL `http://www.jstor.org/stable/2346101`. 2.2.4

[20] W. Lance Bennett and Steven Livingston. The disinformation order: Disruptive communication and the decline of democratic institutions. *European Journal of Communication*, 33(2):122–139, 2018. doi: 10.1177/0267323118760317. 2.1.1

[21] W. Lance Bennett and Steven Livingston. The disinformation order: Disruptive communication and the decline of democratic institutions. *European Journal of Communication*, 33(2):122–139, 2018. doi: 10.1177/0267323118760317. 1.1

[22] Giulia Berlusconi. Do all the pieces matter? assessing the reliability of law enforcement data sources for the network analysis of wire taps. *Global Crime*, 14(1):61–81, 2013. doi: 10.1080/17440572.2012.746940. 1.5.1

[23] Giulia Berlusconi, Francesco Calderoni, Nicola Parolini, Marco Verani, and Carlo Piccardi. Link prediction in criminal networks: A tool for criminal intelligence analysis.

PLoS ONE, 11(4):1–21, 2016. doi: 10.1371/journal.pone.0154244. 1.5.2, 1.5.2

[24] David M. Beskow and Kathleen M. Carley. Bot-hunter: a tiered approach to detecting & characterizing automated activity on Twitter. In *Social, Cultural, and Behavioral Modeling*, volume 10899 of *Lecture Notes in Computer Science*, 2018. 2.1.1, 2.2.5

[25] Janice T. Blane. *Social-Cyber Maneuvers for Analyzing Online Influence Operations*. PhD thesis, Carnegie Mellon University, 2023. **??**, 1.7.1, 3.2.5

[26] Janice T. Blane, Daniele Bellutta, and Kathleen M. Carley. Social-cyber maneuvers analysis during the COVID-19 vaccine initial rollout. *Journal of Medical Internet Research*, 24(3):e34040, 2022. doi: 10.2196/34040. 2.1.1

[27] Ludo Block. The long history of OSINT. *Journal of Intelligence History*, 24(2):95–109, 2024. doi: 10.1080/16161262.2023.2224091. 1.5.3

[28] Stephen P. Borgatti, Kathleen M. Carley, and David Krackhardt. On the robustness of centrality measures under conditions of imperfect data. *Social Networks*, 28(2):124–136, 2006. doi: 10.1016/j.socnet.2005.05.001. 1.1, 1.5.1, 5.1.1, 5.2.1, 5.2.2

[29] Randy Borum. Radicalization into violent extremism I: A review of social science theories. *Journal of Strategic Security*, 4(4):7–36, 2011. doi: 10.5038/1944-0472.4.4.1. 1.5.3

[30] Randy Borum. Radicalization into violent extremism II: A review of conceptual models and empirical research. *Journal of Strategic Security*, 4(4):37–62, 2011. doi: 10.5038/1944-0472.4.4.2. 1.5.3

[31] David Bright, Johan Koskinen, and Aili Malm. Illicit network dynamics: The formation and evolution of a drug trafficking network. *Journal of Quantitative Criminology*, 35(2):237–258, 2019. doi: 10.1007/s10940-018-9379-8. 1.5.2

[32] David A. Broniatowski, Amelia M. Jamison, SiHua Qi, Lulwah AlKulaib, Tao Chen, Adrian Benton, Sandra C. Quinn, and Mark Dredze. Weaponized health communication: Twitter bots and russian trolls amplify the vaccine debate. *American Journal of Public Health*, 108(10):1378–1384, 2018. doi: 10.2105/AJPH.2018.304567. 1.1, 2.1.1

[33] Morgan Burcher and Chad Whelan. Social network analysis as a tool for criminal intelligence: understanding its potential from the perspectives of intelligence analysts. *Trends in Organized Crime*, 21:278–294, 2018. doi: 10.1007/s12117-017-9313-8. 1.5.2, 1.5.2, 6.2.2

[34] Carter T. Butts. Network inference, error, and informant (in)accuracy: a bayesian approach. *Social Networks*, 25(2):103–140, 2003. doi: 10.1016/S0378-8733(02)00038-2. 1.5.1

[35] Isabelle Böhm and Samuel Lolagar. Open source intelligence: Introduction, legal, and ethical considerations. *International Cybersecurity Law Review*, 2(2):317–337, 2021. doi: 10.1365/s43439-021-00042-7. 1.5.3

[36] Kangle Cai, Shan Lu, and Dehong Qiu. Link prediction on dynamic heterogeneous graphs via gnn with multiple attention mechanisms. In *2022 15th International Symposium on Computational Intelligence and Design*, pages 62–65, 2022. doi: 10.1109/ISCID56505.2022.00021. 3.1.1, 3.5.1, 6.2.3

[37] Francesco Calderoni. The structure of drug trafficking mafias: the 'Ndrangheta and cocaine. *Crime, Law and Social Change*, 58:321–349, 2012. doi: 10.1007/s10611-012-9387-9. 1.5.2

[38] Francesco Calderoni, Salvatore Catanese, Pasquale De Meo, Annamaria Ficara, and Giacomo Fiumara. Robust link prediction in criminal networks: A case study of the Sicilian mafia. *Expert Systems with Applications*, 161:113666, 2020. doi: 10.1016/j.eswa.2020.113666. 1.5.2, 5.2.1, 5.6.2

[39] Paolo Campana. The structure of human trafficking: Lifting the bonnet on a Nigerian transnational network. *The British Journal of Criminology*, 56(1):68–86, 2015. doi: 10.1093/bjc/azv027. 1.5.2

[40] Paolo Campana. Out of Africa: The organization of migrant smuggling across the Mediterranean. *European Journal of Criminology*, 15(4):481–502, 2018. doi: 10.1177/1477370817749179. 1.5.2

[41] Paolo Campana and Federico Varese. Studying organized crime networks: Data sources, boundaries and the limits of structural measures. *Social Networks*, 69:149–159, 2022. doi: 10.1016/j.socnet.2020.03.002. 1.5.2, 1.7.1, 1.7.1, 3.1.1, 3.2.5

[42] Kathleen M. Carley. Social cybersecurity: An emerging science. *Computational and Mathematical Organization Theory*, 26(4):365–381, 2020. doi: 10.1007/s10588-020-09322-9. 1.5.3

[43] Kathleen M. Carley, Ju-Sung Lee, and David Krackhardt. Destabilizing networks. *Connections*, 24(3):79–92, 2002. 1.5.2

[44] L. Richard Carley, Jeff Reminga, and Kathleen M. Carley. ORA & NetMapper, 2018. URL `https://sbp-brims.org/2018/proceedings/papers/Demos/ORA%20&%20NetMapper.pdf`. Demonstration at 2018 International Conference on Social Computing, Behavioral-Cultural Modeling, & Prediction and Behavior Representation in Modeling and Simulation. 2.2.5, 2.2.5

[45] Lucia Cavallaro, Annamaria Ficara, Pasquale De Meo, Giacomo Fiumara, Salvatore Catanese, Ovidiu Bagdasar, Wei Song, and Antonio Liotta. Disrupting resilient criminal networks through data analysis: The case of Sicilian mafia. *PLOS ONE*, 15(8):e0236476, 2020. doi: 10.1371/journal.pone.0236476. 1.5.2, **??**, 1.7.1, 3.2.5, 3.4, E.1.2, E.2.2, E.3, E.4, E.5

[46] Ho-Chun Herbert Chang, Emily Chen, Meiqing Zhang, Goran Muric, and Emilio Ferrara. Social bots and social media manipulation in 2020: The year in review, 2021. arXiv:2102.08436v1 [Preprint]. 2.2.1

[47] Jinyuan Chang, Eric D. Kolaczyk, and Qiwei Yao. Estimation of subgraph densities in noisy networks. *Journal of the American Statistical Association*, 117(537):361–374, 2022. doi: 10.1080/01621459.2020.1778482. 4.1.1, 6.1.2

[48] Emily Chen, Ashok Deb, and Emilio Ferrara. #Election2020: the first public Twitter dataset on the 2020 US Presidential election. *Journal of Computational Social Science*, 5:1–18, 2021. doi: 10.1007/s42001-021-00117-9. 2.2.1

[49] Munmun De Choudhury, Winter A. Mason, Jake M. Hofman, and Duncan J. Watts. Inferring relevant social networks from interpersonal communication. In *WWW '10: Proceedings of the 19th International Conference on World Wide Web*, pages 301–310. Association for Computing Machinery, 2010. ISBN 9781605587998. doi: 10.1145/1772690.1772722. 1.5.1

[50] Katherine Clayton, Spencer Blair, Jonathan A. Busam, Samuel Forstner, John Glance, Guy Green, Anna Kawata, Akhila Kovvuri, Jonathan Martin, Evan Morgan, Morgan Sandhu, Rachel Sang, Rachel Scholz-Bright, Austin T. Welch, Andrew G. Wolff, Amanda Zhou, and Brendan Nyhan. Real solutions for fake news? measuring the effectiveness of general warnings and fact-check tags in reducing belief in false stories on social media. *Political Behavior*, 42:1073–1095, 2020. doi: 10.1007/s11109-019-09533-0. 2.1.1

[51] Justin Conrad, Kevin T. Greene, and Brian J. Phillips. Introducing vinsar: Dyadic data on violent non-state actor rivalry. *Journal of Conflict Resolution*, 68(9):1884–1908, 2024. doi: 10.1177/00220027231208708. **??**, 1.7.1, 3.2.5, 3.7, 3.11, E.12, E.13, E.14

[52] Elizabeth Costenbader and Thomas W. Valente. The stability of centrality measures when networks are sampled. *Social Networks*, 25(4):283–307, 2003. doi: 10.1016/S0378-8733(03)00012-1. 1.5.1

[53] Ian Crandell and Gizem Korkmaz. Link prediction in the criminal network of Albuquerque. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 564–567, 2018. doi: 10.1109/ASONAM.2018.8508305. 1.5.2, 1.5.2

[54] Nick Craswell. *Precision at n*, pages 2127–2128. Springer Science+Business Media, 2009. ISBN 9780387399409. doi: 10.1007/978-0-387-39940-9_484. 3.2.3

[55] Stefano Cresci. A decade of social bot detection. *Communications of the ACM*, 63(10):72–83, 2020. doi: 10.1145/3409116. 1.1, 1.5.2, 2.1.1, 2.2.5, 6.4

[56] Brittany Cunningham, James Coldren, Benjamin Carleton, Richard Berk, and Vincent Bauer. Forecasting for police officer safety: A demonstration of concept. *Cambridge Journal of Evidence-Based Policing*, 8:4, 2024. doi: 10.1007/s41887-023-00094-1. 6.2.2

[57] Clayton Allen Davis, Onur Varol, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer. BotOrNot: A system to evaluate social bots. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 273–274, 2016. 2.1.1

[58] Manlio De Domenico, Albert Solé-Ribalta, Emanuele Cozzo, Mikko Kivelä, Yamir Moreno, Mason A. Porter, Sergio Gómez, and Alex Arenas. Mathematical formulation of multilayer networks. *Physical Review X*, 3(4):041022, 2013. doi: 10.1103/PhysRevX.3.041022. 3.1.1, 3.2.1, 3.4.1

[59] Sabine De Moor, Christophe Vandeviver, and Tom Vander Beken. Assessing the missing data problem in criminal network analysis using forensic DNA data. *Social Networks*, 61:99–106, 2020. doi: 10.1016/j.socnet.2019.09.003. 1.5.1

[60] Daryl R. DeFord and Daniel N. Rockmore. A random dot product model for weighted networks, 2016. arXiv:1611.02530 [Preprint]. 4.2.2

[61] Yaneisy Delgado, Bradley S. Price, Paul J. Speaker, and Stephanie L. Stoiloff. Forensic intelligence: Data analytics as the bridge between forensic science and investigation. *Forensic Science International: Synergy*, 3:100162, 2021. doi: 10.1016/j.fsisyn.2021.100162. 1.5

[62] Lina Dencik, Arne Hintz, and Zoe Carey. Prediction, pre-emption and limits to dissent: Social media and big data uses for policing protests in the United Kingdom. *New Media & Society*, 20(4):1433–1450, 2018. 2.4

[63] John Diamond. NSA has massive database of Americans' phone calls. *USA Today*, 10 May 2006. URL `https://usatoday30.usatoday.com/news/washington/2006-05-10-nsa_x.htm`. Accessed 2023-03-16. 1.1

[64] Nick Dietrich and Kristine Eck. Known unknowns: media bias in the reporting of political violence. *International Interactions*, 46(6):1043–1060, 2020. doi: 10.1080/03050629.2020.1814758. 1.7.1, 3.2.5

[65] Konstantinos Drakos and Andreas Gofas. The devil you know but are afraid to face: Underreporting bias and its distorting effects on the study of terrorism. *Journal of Conflict Resolution*, 50(5):714–735, 2006. doi: 10.1177/0022002706291051. 1.7.1, 3.2.5

[66] Paul A. C. Duijn and Peter P. H. M. Klerks. *Social Network Analysis Applied to Criminal Networks: Recent Developments in Dutch Law Enforcement*, pages 121–159. Lecture Notes in Social Networks. Springer International Publishing, 2014. ISBN 9783319041469. doi: doi.org/10.1007/978-3-319-04147-6_6. 1.5.2

[67] Daniel M. Dunlavy, Tamara G. Kolda, and Evrim Acar. Temporal link prediction using matrix and tensor factorizations. *ACM Transactions on Knowledge Discovery from Data*, 5(2):1–27, 2011. doi: 10.1145/1921632.1921636. 3.2.1

[68] Christina Durón, Yuan Pan, David H. Gutmann, Johanna Hardin, and Ami Radunskaya. Variability of betweenness centrality and its effect on identifying essential genes. *Bulletin of Mathematical Biology*, 81(9):3655–3673, 2019. doi: 10.1007/s11538-018-0526-z. 4.1.1, 6.1.2

[69] Facebook. How Facebook's third-party fact-checking program works. Facebook Journalism Project, 2021. URL `https://www.facebook.com/journalismproject/programs/third-party-fact-checking/how-it-works`. Accessed 2022-03-10. 2.4, 2.5.2, 6.2.3

[70] Facebook. The state of influence operations 2017-2020. Technical report, Facebook, 2021. Accessed 2024-03-27. 2.1.1, 2.3.7, 2.4, 6.1.1

[71] Lucia Falzon, Eric Quintane, John Dunn, and Garry Robins. Embedding time in positions: Temporal measures of centrality for social network analysis. *Social Networks*, 54:168–178, 2018. doi: 10.1016/j.socnet.2018.02.002. 5.3.1

[72] Ling Fang, Haiyang Fang, Yanfang Tian, Tinghong Yang, and Jing Zhao. The alliance relationship analysis of international terrorist organizations with link prediction. *Physica A: Statistical Mechanics and its Applications*, 482:573–584, 2017. doi: 10.1016/j.physa.2017.04.068. 1.5.2

[73] Katherine Faust and George E. Tita. Social networks and crime: Pitfalls and promises for advancing the field. *Annual Review of Criminology*, 2:99–122, 2019. doi: 10.1146/annurev-criminol-011518-024701. 1.5.2

[74] Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer, and Alessandro Flammini. The rise of social bots. *Communications of the ACM*, 59(7):96–104, 2016. 2.2.5

[75] Emilio Ferrara, Herbert Chang, Emily Chen, Goran Muric, and Jaimin Patel. Characterizing social media manipulation in the 2020 U.S. presidential election. *First Monday*, 25 (11), 2020. 2.2.1

[76] Annamaria Ficara, Lucia Cavallaro, Francesco Curreri, Giacomo Fiumara, Pasquale De Meo, Ovidiu Bagdasar, Wei Song, and Antonio Liotta. Criminal networks analysis in missing data scenarios through graph distances. *PLOS ONE*, 16(8):e0255067, 2021. doi: 10.1371/journal.pone.0255067. 1.5.1, 1.5.2, **??**, 1.7.1, 3.2.5, 3.4, 5.2.1, E.1.2, E.2.2, E.3, E.4, E.5

[77] Annamaria Ficara, Giacomo Fiumara, Salvatore Catanese, Pasquale De Meo, and Xiaoyang Liu. The whole is greater than the sum of the parts: A multilayer approach on criminal networks. *Future Internet*, 14(5):123, 2022. doi: 10.3390/fi14050123. 1.5.2

[78] Terrill L. Frantz and Kathleen M. Carley. Reporting a network's most-central actor with a confidence level. *Computational Mathematical Organization Theory*, 23(2):301–312, 2017. doi: 10.1007/s10588-016-9229-x. 1.1, 4.1.1, 4.2.2

[79] Terrill L. Frantz, Marcelo Cataldo, and Kathleen M. Carley. Robustness of centrality measures under uncertainty: Examining the role of network topology. *Computational and Mathematical Organization Theory*, 15(4):303–328, 2009. doi: 10.1007/s10588-009-9063-5. 1.5.1

[80] Sheng Gao, Ludovic Denoyer, Patrick Gallinari, and Jun Guo. Probabilistic latent tensor factorization model for link pattern prediction in multi-relational networks. *The Journal of China Universities of Posts and Telecommunications*, 19(2):172–181, 2012. doi: 10.1016/S1005-8885(11)60425-1. 3.2.1

[81] Diego Garlaschelli. The weighted random graph model. *New Journal of Physics*, 11: 073005, 2009. doi: 10.1088/1367-2630/11/7/073005. 3.2.4

[82] Amir Ghasemian, Homa Hosseinmardi, Aram Galstyan, Edoardo M. Airoldi, and Aaron Clauset. Stacking models for nearly optimal link prediction in complex networks. *Proceedings of the National Academy of Sciences*, 117(38):23393–23400, 2020. doi: 10.1073/pnas.1914950117. 1.5.2, 3.4.1, 3.5.1, 6.3, 6.4

[83] Michael Glassman and Min Ju Kang. Intelligence in the internet age: The emergence and evolution of open source intelligence (OSINT). *Computers in Human Behavior*, 28(2): 673–682, 2012. doi: https://doi.org/10.1016/j.chb.2011.11.014. 1.1, 1.5.3

[84] Carlos Henrique Gomes Ferreira, Fabricio Murai, Ana P. C. Silva, Martino Trevisan, Luca Vassio, Idilio Drago, Marco Mellia, and Jussara M. Almeida. On network backbone extraction for modeling online collective behavior. *PLOS ONE*, 17(9):1–36, 2022. doi: 10.1371/journal.pone.0274218. 1.5.1

[85] Martin C. Goossen and Srikanth Paruchuri. Measurement errors and estimation biases with incomplete social networks: replication studies on intra-firm inventor network analysis. *Research Policy*, 51(1):104404, 2022. doi: 10.1016/j.respol.2021.104404. 1.5.1

[86] Alden Green and Cosma Rohilla Shalizi. Bootstrapping exchangeable random graphs. *Electronic Journal of Statistics*, 16:1058–1095, 2022. doi: 10.1214/21-EJS1896. 4.1.1, 4.5.2, 6.1.2

[87] Alden Green and Cosma Rohilla Shalizi. Bootstrapping exchangeable random graphs. *Electronic Journal of Statistics*, 16:1058–1095, 2022. doi: 10.1214/21-EJS1896. 4.1.1

[88] W. H. Greene. *Econometric Analysis*. Pearson, Boston, 7 edition, 2012. ISBN 9780131395381. 2.2.4

[89] Nir Grinberg, Kenneth Joseph, Lisa Friedland, Briony Swire-Thompson, and David Lazer. Fake news on Twitter during the 2016 U.S. presidential election. *Science*, 363(6425):374–378, 2019. 2.1.1

[90] Nir Grinberg, Kenneth Joseph, Lisa Friedland, Briony Swire-Thompson, and David Lazer. Public replication package for fake news on Twitter. Zenodo, 2019. URL `https://doi.org/10.5281/zenodo.2651401`. 2.2.5

[91] Roger Guimerà and Marta Sales-Pardo. Missing and spurious interactions and the reconstruction of complex networks. *Proceedings of the National Academy of Sciences*, 106 (52):22073–22078, 2009. doi: 10.1073/pnas.0908366106. 3.1.1, 3.4.2, 3.5.2, 5.2.1, 5.6.2

[92] Timothy C. Haas and Sam M. Ferreira. Federated databases and actionable intelligence: using social network analysis to disrupt transnational wildlife trafficking criminal networks. *Security Informatics*, 4(1):2, 2015. doi: 10.1186/s13388-015-0018-8. 1.5.2

[93] Mark S. Handcock and Krista Gile. Modeling social networks with sampled or missing data. Working Paper 75, University of Washington Center for Statistics and the Social Sciences, April 2007. URL `https://csss.uw.edu/files/working-papers/2007/wp75.pdf`. Accessed 2024-04-14. 1.4

[94] Walter R. Harper and Douglas H. Harris. The application of link analysis to police intelligence. *Human Factors*, 17(2):157–164, 1975. doi: 10.1177/001872087501700206. 1.5.2, 1.5.2

[95] Xie He, Amir Ghasemian, Eun Lee, Alice C. Schwarze, Aaron Clauset, and Peter J. Mucha. Link prediction accuracy on real-world networks under non-uniform missing-edge patterns. *PLoS ONE*, 19(7):e0306883, 2024. doi: 10.1371/journal.pone.0306883. 3.5.1, 3.5.2

[96] Nicholas A. Heard, David J. Weston, Kiriaki Platanioti, and David J. Hand. Bayesian anomaly detection methods for social networks. *The Annals of Applied Statistics*, 4(2): 645–662, 2010. doi: 10.1214/10-AOAS329. 6.4

[97] John S. Hollywood, Michael J. D. Vermeer, Dulani Woods, Sean E. Goodison, and Brian A. Jackson. Using social media and social network analysis in law enforcement: Creating a research agenda, including business cases, protections, and technology needs. Technical report, RAND Corporation, 2018. 1.1

[98] Max Houck and Gregory D. Smith. The forensic mindset: art and crime. *Forensic Science International: Synergy*, 3:100143, 2021. doi: 10.1016/j.fsisyn.2021.100143. 1.1, 1.5, 6.1

[99] Mark Huisman and Robert M. Krause. *Imputation of Missing Network Data*, pages 1–10. Springer Science+Business Media, 2017. ISBN 9781461471639. doi: 10.1007/978-1-4614-7163-9_394-1. 1.4

[100] Yuriy Hulovatyy, Ryan W. Solava, and Tijana Milenković. Revealing missing parts of the interactome via link prediction. *PLOS ONE*, 9(3):e90073, 2014. doi: 10.1371/journal.pone.0090073. 5.2.1, 5.6.2

[101] Bart Husslage, Peter Borm, Twan Burg, Herbert Hamers, and Roy Lindelauf. Ranking terrorists in networks: A sensitivity analysis of Al Qaeda's 9/11 attack. *Social Networks*, 42:1–7, 2015. doi: 10.1016/j.socnet.2015.02.003. 1.5.2

[102] Jacopo Iacovacci and Ginestra Bianconi. Extracting information from multiplex networks. *Chaos*, 26:065306, 2016. doi: 10.1063/1.4953161. 3.1.1

[103] Paul Jaccard. The distribution of the flora in the Alpine zone. *New Phytologist*, 11(2): 37–50, 1912. doi: 10.1111/j.1469-8137.1912.tb05611.x. 5.3.2, 5.2c, 5.2d

[104] Farnaz Jahanbakhsh, Amy X. Zhang, Adam J. Berinsky, Gordon Pennycook, David G. Rand, and David R. Karger. Exploring lightweight interventions at posting time to reduce the sharing of misinformation on social media. *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW1), 2021. 2.1.1

[105] Jennifer A. Johnson, John David Reitzel, Bryan F. Norwood, David M. McCoy, D. Brian Cummings, and Renee R. Tate. Social network analysis: A systematic approach for investigating. *Law Enforcement Bulletin*, 5 March 2013. URL `https://leb.fbi.gov/articles/featured-articles/social-network-analysis-a-systematic-approach-for-investigating`. Accessed 2023-03-16. 1.1

[106] Marc Owen Jones. Propaganda, fake news, and fake trends: The weaponization of Twitter bots in the gulf crisis. *International Journal of Communication*, 13:1389–1415, 2019. 2.1.1, 6.1.1

[107] Terrence D. Jorgensen, K. Jean Forney, Jeffrey A. Hall, and Steven M. Giles. Using modern methods for missing data analysis with the social relations model: A bridge to social network analysis. *Social Networks*, 54:26–40, 2018. doi: 10.1016/j.socnet.2017.11.002. 1.4

[108] Orsolya Kardos, András London, and Tamás Vinkó. Stability of network centrality measures: a numerical study. *Social Network Analysis and Mining*, 10:80, 2020. doi: 10.1007/s13278-020-00693-0. 5.1.1, 5.3.1

[109] Michael Kenney and Stephen Coulthart. *The Methodological Challenges of Extracting Dark Networks: Minimizing False Positives through Ethnography*, pages 52–70. Structural Analysis in the Social Sciences. Cambridge University Press, 2015. ISBN 9781316212639. doi: 10.1017/CBO9781316212639.005. 1.5.2

[110] Mohammad Reza Keyvanpour, Mohammad Moradi, and Faranak Hasanzadeh. *Digital*

Forensics 2.0: A Review on Social Networks Forensics, volume 555 of *Studies in Computational Intelligence*, pages 17–46. Springer, 2014. ISBN 978-3-319-05884-9. doi: 10.1007/978-3-319-05885-6_2. 6.2.2

[111] Yoonsang Kim, Jidong Huang, and Sherry Emery. Garbage in, garbage out: Data collection, quality assessment and reporting standards for social media data use in health research, infodemiology and digital disease detection. *Journal of Medical Internet Research*, 18(2):e41, 2016. doi: 10.2196/jmir.4738. 5.4.1, 6.2.3

[112] Tamara G. Kolda and Brett W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009. doi: 10.1137/07070111X. 3.2.1, **??**, **??**

[113] Johan H. Koskinen, Garry L. Robins, Peng Wang, and Philippa E. Pattison. Bayesian analysis for partially observed network data, missing ties, attributes and actors. *Social Networks*, 35:514–527, 2013. doi: 10.1016/j.socnet.2013.07.003. 1.4

[114] Gueorgi Kossinets. Effects of missing data in social networks. *Social Networks*, 28(3): 247–268, 2006. doi: 10.1016/j.socnet.2005.07.002. 1.5.1

[115] Robert W. Krause, Mark Huisman, Christian Steglich, and Tom A. B. Snijders. Missing network data: A comparison of different imputation methods. In *Proceedings of the 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 159–163. IEEE, 2018. ISBN 9781538660515. doi: 10.1109/ASONAM.2018.8508716. 5.2.1

[116] Robert W. Krause, Mark Huisman, Christian Steglich, and Tom Snijders. Missing data in cross-sectional networks – an extensive comparison of missing data treatment methods. *Social Networks*, 62:99–112, 2020. doi: j.socnet.2020.02.004. 1.4, 5.2.1

[117] Valdis E. Krebs. Mapping networks of terrorist cells. *Connections*, 24(3):43–52, 2002. 1.5.2, 1.5.2

[118] Ajay Kumar, Shashank Sheshar Singh, Kuldeep Singh, and Bhaskar Biswas. Link prediction techniques, applications, and performance: A survey. *Physica A: Statistical Mechanics and its Applications*, 553:124289, 2020. doi: 10.1016/j.physa.2020.124289. 1.5.2, 1.5.2, 3.1.1, 3.2.2, 3.2.3, 5.2.1

[119] Sumeet Kumar. *Social Media Analytics for Stance Mining: A Multi-Modal Approach with Weak Supervision*. Ph.d. dissertation, Carnegie Mellon University, Pittsburgh, PA, USA, 2020. Section 4.3.1. 2.2.5

[120] T. Jaya Lakshmi and S. Durga Bhavani. Link prediction in temporal heterogeneous networks. In *Intelligence and Security Informatics*, pages 83–98, 2017. doi: 10.1007/978-3-319-57463-9_6. 3.1.1, 6.2.3

[121] Sangho Lee and Jong Kim. Early filtering of ephemeral malicious accounts on Twitter. *Computer Communications*, 54(C):48–57, 2014. doi: 10.1016/j.comcom.2014.08.006. 2.4

[122] Kristina Lerman, Rumi Ghosh, and Jeon Hyung Kang. Centrality metric for dynamic networks. In *Proceedings of the Eighth Workshop on Mining and Learning with Graphs*, pages 70–77, 2010. doi: 10.1145/1830252.1830262. 5.3.1

[123] Keith D. Levin and Elizaveta Levina. Bootstrapping networks with latent space structure, 2021. arXiv:1907.10821v2 [Preprint]. 4.1.1, 4.2.2, 4.2.3, 4.4, 4.5.1, 4.5.2, 6.1.2

[124] Rachel Levinson-Waldman and Mary Pat Dwyer. LAPD documents show what one social media surveillance firm promises police. *Brennan Center for Justice*, 2021. URL https://www.brennancenter.org/our-work/analysis-opinion/lapd-documents-show-what-one-social-media-surveillance-firm-promises. Accessed 2024-03-30. 1.1, 1.5.3, 1.5.3, 6.2.2

[125] Ryan Lichtnwalter and Nitesh V. Chawla. Link prediction: Fair and effective evaluation. In *Proceedings of the 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 376–383. IEEE Computer Society, 2012. ISBN 9780769547992. doi: 10.1109/ASONAM.2012.68. 3.2.3

[126] Roderick J. A. Little and Donald B. Rubin. *Statistical analysis with missing data*. Wiley, 1987. ISBN 0471802549. 1.5.1, 3.1.1, 3.2.4, 3.2.5, 3.2.5, 3.5.2, 4.1.2, 4.2.1, 4.5.1

[127] Linyuan Lü and Tao Zhou. Link prediction in weighted networks: The role of weak ties. *Europhysics Letters*, 89(1):18001, 2010. doi: 10.1209/0295-5075/89/18001. 3.4.1, 3.5.2

[128] Linyuan Lü and Tao Zhou. Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and its Applications*, 390(6):1150–1170, 2011. doi: https://doi.org/10.1016/j.physa.2010.11.027. 1.5.2, 3.1.1, 5.2.1

[129] Linyuan Lü, Liming Pan, Tao Zhou, Yi-Cheng Zhang, and H. Eugene Stanley. Toward link predictability of complex networks. *Proceedings of the National Academy of Sciences*, 112(8):2325–2330, 2015. doi: 10.1073/pnas.1424644112. 3.2.2, 4.1.1, 4.2.2

[130] Thomas Magelinski, Lynnette Hui Xian Ng, and Kathleen M. Carley. A synchronized action framework for detection of coordination on social media. *Journal of Online Trust and Safety*, 1(2), 2022. doi: 10.54501/jots.v1i2.30. 2.1.1

[131] Sean McKeown, David Maxwell, Leif Azzopardi, and William Bradley Glisson. Investigating people: a qualitative analysis of the search behaviours of open-source intelligence analysts. In *Proceedings of the 5th Information Interaction in Context Symposium (IIiX '14)*, pages 175–184, 2014. ISBN 9781450329767. doi: 10.1145/2637002.2637023. 1.5.3

[132] Media Bias/Fact Check. Conspiracy-pseudoscience. Media Bias/Fact Check, 2020. URL https://mediabiasfactcheck.com/filtered-search/?bias=Conspiracy-Pseudoscience. 2.2.5

[133] Media Bias/Fact Check. Questionable sources. Media Bias/Fact Check, 2020. URL https://mediabiasfactcheck.com/filtered-search/?bias=Questionable%20Sources. 2.2.5

[134] Cesare Miglioli and Jordan Awan. Incomplete U-statistics of equireplicate designs: Berry-Esseen bound and efficient construction, 2025. arXiv:2510.20755v3 [Preprint]. 4.5.1

[135] Bowman H. Miller. Open source intelligence (OSINT): An oxymoron? *International Journal of Intelligence and CounterIntelligence*, 31(4):702–719, 2018. doi: 10.1080/08850607.2018.1492826. 1.5.3, 1.5.3

[136] Ron Milo, Shalev Itzkovitz, Nadav Kashtan, Reuven Levitt, Shai Shen-Orr, Inbal Ayzenshtat, Michal Sheffer, and Uri Alon. Superfamilies of evolved and designed networks. *Science*, 303(5663):1538–1542, 2004. doi: 10.1126/science.1089167. 4.1.1, 6.1.2

[137] Rachel Minster, Arvind K. Saibaba, and Misha E. Kilmer. Randomized algorithms for low-rank tensor decompositions in the Tucker format. *SIAM Journal on Mathematics of Data Science*, 2(1):189–215, 2020. doi: 10.1137/19M1261043. 3.2.1

[138] Wendy W. Moe and David A. Schweidel. *Social Media Intelligence*. Cambridge University Press, 2014. doi: 10.1017/CBO9781139381338. 1.5.3, 1.5.3

[139] Carlo Morselli, Cynthia Giguère, and Katia Petit. The efficiency/security trade-off in criminal networks. *Social Networks*, 29(1):143–153, 2007. doi: 10.1016/j.socnet.2006.05.001. 1.5.2

[140] Fred Morstatter, Jürgen Pfeffer, Huan Liu, and Kathleen M. Carley. Is the sample good enough? comparing data from Twitter's streaming API with Twitter's firehose. In *Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media*, pages 400–408, 2013. ISBN 9781577356103. doi: 10.1609/icwsm.v7i1.14401. 1.1, 1.5.1, 1.7.1, 5.1.1, 5.2.1

[141] Elahe Nasiri, Kamal Berahmand, and Yuefeng Li. A new link prediction in multiplex networks using topologically biased random walks. *Chaos, Solitons & Fractals*, 151:111230, 2021. doi: 10.1016/j.chaos.2021.111230. 3.2.2, 3.2.5

[142] National Academies of Sciences, Engineering, and Medicine. *A Decadal Survey of the Social and Behavioral Sciences: A Research Agenda for Advancing Intelligence Analysis*. National Academies Press, 2019. ISBN 9780309487610. 1.5.3

[143] Meredith Neville-Shepard. Masks and emasculation: Populist crisis rhetoric and the 2020 presidential election. *American Behavioral Scientist*, 68(1), 2021. doi: 10.1177/00027642211011223. 2.2.5

[144] M. E. J. Newman. Clustering and preferential attachment in growing networks. *Physical Review E*, 64(2):025102, 2001. doi: 10.1103/PhysRevE.64.025102. 3.2.2

[145] Lynnette Hui Xian Ng and Kathleen M. Carley. A combined synchronization index for evaluating collective action social media. *Applied Network Science*, 8:1, 2023. doi: 10.1007/s41109-022-00526-3. 2.2.5

[146] Lynnette Hui Xian Ng, Dawn C. Robertson, and Kathleen M. Carley. Stabilizing a supervised bot detection algorithm: How much data is needed for consistent predictions? *Online Social Networks and Media*, 28:100198, 2022. 2.2.5

[147] Ben Nimmo, C. Shawn Eib, and Léa Ronzaud. Operation Naval Gazing. Technical report, Graphika, 2020. Accessed 2024-03-27. 2.1.1, 2.2.3, 2.3.7, 6.1.1

[148] Kate O'Keeffe and Cameron McWhirter. New Orleans DA fights 'terrorism' on streets with AI spycraft. *The Wall Street Journal*, 17 September 2023. URL `https://www.wsj.com/us-news/law/new-orleans-da-fights-terrorism-on-streets-with-ai-spycraft-87f74055`. Accessed 2024-03-30. 1.1, 1.5.3, 1.5.3

[149] David Omand, Jamie Bartlett, and Carl Miller. Introducing social media intelligence (SOCMINT). *Intelligence and National Security*, 27(6):801–823, 2012. doi: 10.1080/02684527.2012.716965. 1.5.3, 1.5.3

[150] Alper Ozcan and Sule Gunduz Oguducu. Link prediction in evolving heterogeneous networks using the NARX neural networks. *Knowledge and Information Systems*, 55(2): 333–360, 2018. doi: 10.1007/s10115-017-1073-x. 3.1.1, 3.5.1, 6.2.3

[151] Diogo Pacheco, Pik-Mai Hui, Christopher Torres-Lugo, Bao Tran Truong, Alessandro Flammini, and Filippo Menczer. Uncovering coordinated networks on social media: Methods and case studies. In *Proceedings of the Fifteenth International AAAI Conference on Web and Social Media*, pages 455–466, 2021. doi: 10.1609/icwsm.v15i1.18075. 2.1.1

[152] Liming Pan, Tao Zhou, Linyuan Lü, and Chin-Kun Hu. Predicting missing links and identifying spurious links via likelihood analysis. *Scientific Reports*, 6(1):22955, 2016. doi: 10.1038/srep22955. 3.1.1, 3.3.1

[153] L. E. Papke and J. M. Woolridge. Econometric methods for fractional response variables with an application to 401(k) plan participation rates. *Journal of Applied Econometrics*, 11(6):619–632, 1996. doi: 10.1002/(SICI)1099-1255(199611)11:6<619::AID-JAE418>3.0.CO;2-1. 2.2.4

[154] Deven Parekh, Amarnath Amarasingam, Lorne Dawson, and Derek Ruths. Studying jihadists on social media: A critique of data collection methodologies. *Perspectives on Terrorism*, 12(3):5–23, 2018. URL `https://www.jstor.org/stable/26453132`. 5.4.1, 5.5.1, 5.6.2, 6.2.3

[155] Ratha Pech, Dong Hao, Liming Pan, Hong Cheng, and Tao Zhou. Link prediction via matrix completion. *Europhysics Letters*, 117(3):38002, 2017. doi: 10.1209/0295-5075/117/38002. 3.2.2

[156] Leto Peel, Tiago P. Peixoto, and Manlio De Domenico. Statistical inference links data and theory in network science. *Nature Communications*, 13:6794, 2022. doi: 10.1038/s41467-022-34267-9. 1.1, 6.2.1, 6.4

[157] Joshua J. Peterson. Appropriate factors to consider when assessing analytic confidence in intelligence analysis. Master's thesis, Mercyhurst College, 2008. 5.2.4

[158] Elizabeth Phillips, Jason R. C. Nurse, Michael Goldsmith, and Sadie Creese. Extracting social structure from darkweb forums. In *Fifth International Conference on Social Media Technologies, Communication, and Informatics (SOTICS 2015)*, pages 97–102, 2015. ISBN 9781612084435. 1.5.2, 1.6, 3.2.5

[159] Xiaochen Pi, Longkun Tang, and Xiangzhong Chen. A directed weighted scale-free network model with an adaptive evolution mechanism. *Physica A: Statistical Mechanics and its Applications*, 572:125897, 2021. doi: 10.1016/j.physa.2021.125897. 3.2.4

[160] PolitiFact. PolitiFact's guide to fake news websites and what they peddle. PolitiFact, 2017. URL `https://www.politifact.com/article/2017/apr/20/politifacts-guide-fake-news-websites-and-what-they/`. Accessed

2022-03-09. 2.2.5

[161] Hui Qu, Wei Chen, and Kuo Chi. Missing link prediction and spurious link detection based on attractive force and community. *Science Progress*, 104(2):00368504211018558, 2021. doi: 10.1177/00368504211018558. 3.1.1, 3.4.2, 3.5.2

[162] Sebastian Raimondo and Manlio De Domenico. Measuring topological descriptors of complex networks under uncertainty. *Physical Review E*, 103(2):022311, 2021. doi: 10.1103/PhysRevE.103.022311. 4.1.1, 6.1.2

[163] Esmeralda A. Ramalho, Joaquim J. S. Ramalho, and José M. R. Murteira. Alternative estimating and testing empirical strategies for fractional regression models. *Journal of Economic Surveys*, 25(1):19–68, 2011. doi: 10.1111/j.1467-6419.2009.00602.x. 2.2.4

[164] Julio C. S. Reis, André Correia, Fabrício Murai, Adriano Veloso, and Fabrício Benevenuto. Supervised learning for fake news detection. *IEEE Intelligent Systems*, 34(2):76–81, 2019. 2.1.1

[165] Mark A. Richards, James A. Scheer, and William A. Holm. *Principles of Modern Radar, Volume I - Basic Principles*. SciTech Publishing, 2010. ISBN 9781891121524. 2.2.3

[166] Donald B. Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976. doi: 10.1093/biomet/63.3.581. 1.5.1, 3.1.1, 3.2.4, 3.2.5, 3.2.5, 3.5.2, 4.1.2, 4.2.1, 4.5.1

[167] Zeynab Samei and Mahdi Jalili. Discovering spurious links in multiplex networks based on interlayer relevance. *Journal of Complex Networks*, 7(5):641–658, 2019. doi: 10.1093/comnet/cnz007. 1.5.1, 3.1.1

[168] David Savage, Xiuzhen Zhang, Xinghuo Yu, Pauline Chou, and Qingmai Wang. Anomaly detection in online social networks. *Social Networks*, 39:62–70, 2014. doi: 10.1016/j.socnet.2014.05.002. 6.4

[169] Joseph L. Schafer and John W. Graham. Missing data: Our view of the state of the art. *Psychological Methods*, 7(2):147–177, 2002. doi: 10.1037//1082-989X.7.2.147. 1.5.1, 3.1.1, 3.2.4, 3.2.5, 3.2.5, 3.5.2, 4.1.2, 4.2.1, 4.5.1

[170] Edward R. Scheinerman and Kimberly Tucker. Modeling graphs using dot product representations. *Computational Statistics*, 25(1):1–16, 2010. doi: 10.1007/s00180-009-0158-8. 4.1.1, 4.2.2, 6.1.2

[171] Joey Scott. LAPD is using Israeli surveillance software that can track your phone and social media. *Knock LA*, 27 November 2023. URL `https://knock-la.com/lapd-is-using-israeli-surveillance-software-that-can-track-your-phone-and-social-media/`. Accessed 2024-03-30. 1.1, 1.5.3, 1.5.3

[172] Santiago Segarra and Alejandro Ribeiro. A stable betweenness centrality measure in networks. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3859–3863, 2014. doi: 10.1109/ICASSP.2014.6854324. 5.3.1, 5.6.2

[173] Santiago Segarra and Alejandro Ribeiro. Stability and continuity of centrality measures in weighted graphs. *IEEE Transactions on Signal Processing*, 64(3):543–555, 2016. doi:

10.1109/TSP.2015.2486740. 5.3.2, 5.3.3, 5.2b

[174] M. Ángeles Serrano, Marián Boguña, and Alessandro Vespignani. Extracting the multiscale backbone of complex weighted networks. *Proceedings of the National Academy of Sciences*, 106(16):6483–6488, 2009. doi: 10.1073/pnas.0808904106. 1.5.1

[175] Karishma Sharma, Emilio Ferrara, and Yan Liu. Characterizing online engagement with disinformation and conspiracies in the 2020 U.S. presidential election. In *Proceedings of the Sixteenth International AAAI Conference on Web and Social Media*, pages 908–919, 2022. 2.2.1

[176] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media: A data mining perspective. *SIGKDD Exploration Newsletter*, 19(1):22–36, 2017. 2.1.1

[177] Jeffrey A. Smith and James Moody. Structural effects of network sampling coverage I: Nodes missing at random. *Social Networks*, 35(4):652–668, 2013. doi: 10.1016/j.socnet.2013.09.003. 1.5.1

[178] Steven T. Smith, Edward K. Kao, Erika D. Mackin, Danelle C. Shah, Olga Simek, and Donald B. Rubin. Automatic detection of influential actors in disinformation networks. *Proceedings of the National Academy of Sciences*, 118(4):e2011216118, 2021. doi: 10.1073/pnas.2011216118. 2.1.1

[179] H. Francis Song and Xiao-Jing Wang. Simple, distance-dependent formulation of the watts-strogatz model for directed and undirected small-world networks. *Physical Review E*, 90(6):062801, 2014. doi: 10.1103/PhysRevE.90.062801. 3.2.4

[180] Malcolm K. Sparrow. The application of network analysis to criminal intelligence: An assessment of the prospects. *Social Networks*, 13(3):251–274, 1991. doi: 10.1016/0378-8733(91)90008-H. 1.1, 1.5.2, 1.5.2

[181] Francesca Spezzano, V. S. Subrahmanian, and Aaron Mannes. Reshaping terrorist networks. *Communications of the ACM*, 57(8):60–69, 2014. doi: 10.1145/2632661.2632664. 1.5.2

[182] Stephan Spiegel, Jan Clausen, Sahin Albayrak, and Jérôme Kunegis. Link prediction on evolving data using tensor factorization. In *New Frontiers in Applied Data Mining*, pages 100–110, 2012. doi: 10.1007/978-3-642-28320-8_9. 3.2.1

[183] Andrew Staniforth. *Police Use of Open Source Intelligence: The Longer Arm of Law*, pages 21–31. Advanced Sciences and Technologies for Security Applications. Springer International Publishing, 2016. ISBN 9783319476711. doi: 10.1007/978-3-319-47671-1_3. 1.1

[184] V. S. Subrahmanian, editor. *Handbook of Computational Approaches to Counterterrorism*. Springer, 1 edition, 2012. doi: 9781461453109. 1.5.2

[185] Richard Takacs and Ian McCulloh. Dormant bots in social media: Twitter and the 2018 U.S. Senate election. In *2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 796–800, 2019. ISBN 9781450368681. doi: 10.1145/3341161.3343852. 2.1.1, 6.1.1

[186] Dane Taylor, Sean A. Myers, Aaron Clauset, Mason A. Porter, and Peter J. Mucha. Eigenvector-based centrality measures for temporal networks. *Multiscale Modeling & Simulation*, 15(1):537–574, 2017. doi: 10.1137/16M1066142. 5.3.1

[187] Thomas Thorne and Michael P. H. Stumpf. Generating confidence intervals on biological networks. *BMC Bioinformatics*, 8(1):467, 2007. doi: 10.1186/1471-2105-8-467. 4.1.1, 6.1.2

[188] Ramine Tinati, Les Carr, Wendy Hall, and Johnny Bentwood. Scale free: Twitter's retweet network structure, 2012. URL `https://eprints.soton.ac.uk/344807/1/Network_Science_2012_-_Scale-Free_Twitter_27s_Retweet_Network_Structure_V2.pdf`. NetSci 2012: Conference on Network Science. [Poster]. 4.5.2, 5.5.2

[189] Alex S. O. Toledo, Laura C. Carpi, Allbens P. F. Atman, and Antônio P. B. Scarpelli. Multiplex key roles to disrupt criminal networks. *Social Network Analysis and Mining*, 13(1):98, 2023. doi: 10.1007/s13278-023-01101-z. **??**, 1.7.1, 3.2.5, 3.3, E.1.1, E.1, E.2

[190] Hanghang Tong, Spiros Papadimitriou, Jimeng Sun, Philip S. Yu, and Christos Faloutsos. Colibri: Fast mining of large static and dynamic graphs. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 686–694, 2008. doi: 10.1145/1401890.1401973. 3.2.1, **??**

[191] V. A. Traag, L. Waltman, and N. J. van Eck. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific Reports*, 9:5233, 2019. doi: 10.1038/s41598-019-41695-z. 1.1, 1.6, 6.1

[192] Daniel Trottier. Open source intelligence, social media and law enforcement: Visions, constraints and critiques. *European Journal of Cultural Studies*, 18(4–5):530–547, 2015. doi: 10.1177/1367549415577396. 1.5.3, 1.5.3

[193] Twitter. Serving healthy conversation. Twitter Blog, 2018. URL `https://blog.twitter.com/official/en_us/topics/product/2018/Serving_Healthy_Conversation.html`. Accessed 2020-03-10. 2.4, 2.5.2, 6.2.3

[194] Twitter. Academic research access. Twitter Developer Platform, 2023. URL `https://developer.twitter.com/en/products/twitter-api/academic-research`. Accessed 2023-04-17. 1.7.1, 3.2.5, C.1.1

[195] Twitter. Stream tweets in real-time. Twitter Developer Platform, 2023. URL `https://developer.twitter.com/en/docs/tutorials/stream-tweets-in-real-time`. Accessed 2023-04-17. 1.7.1, 1.7.1, 3.2.5

[196] Twitter Safety. Information operations directed at Hong Kong. Technical report, Twitter, 2019. Accessed 2024-03-27. 2.1.1, 2.3.7, 2.4, 6.1.1

[197] University of Arizona Artificial Intelligence Laboratory. Islamic Network, 2012. URL `https://dibbs.ai.arizona.edu/dibbs/azsecure-forums-darkweb/IslamicNetwork.zip`. Web forum data. 1.6, **??**, 1.7.1, 3.2.5, 3.8, C.1.2, E.2.3, E.15, E.16, E.17

[198] Joshua Uyheng, Thomas Magelinski, Ramon Villa-Cox, Christine Sowa, and Kathleen M. Carley. Interoperable pipelines for social cyber-security: Assessing Twitter information operations during NATO Trident Juncture 2018. *Computational and Mathematical Organization Theory*, 26(4):465–483, 2020. 2.1.1, 2.4, 2.5.2, 6.1.1, 6.2.3

[199] George Valkanas, Ioannis Katakis, and Dimitrios Gunopulos. Mining Twitter data with resource constraints. In *2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, pages 157–164. IEEE Computer Society, 2014. ISBN 9781479941438. doi: 10.1109/WI-IAT.2014.29. 1.1, 1.5.1, 5.1.1, 5.2.1

[200] Karl M. van Meter. Terrorists/liberators: Researching and dealing with adversary social networks. *Connections*, 24(3):66–78, 2001. 1.5.2

[201] Soroush Vosoughi, Deb Roy, and Sinan Aral. The spread of true and false news online. *Science*, 359(6380):1146–1151, 2018. 2.4

[202] Emily K. Vraga, Leticia Bode, and Melissa Tully. Creating news literacy messages to enhance expert corrections of misinformation on Twitter. *Communication Research*, 49 (2):245–267, 2022. doi: 10.1177/0093650219898094. 2.1.1

[203] Dan J. Wang, Xiaolin Shi, Daniel A. McFarland, and Jure Leskovec. Measurement error in network data: A re-classification. *Social Networks*, 34(4):396–409, 2012. doi: 10.1016/j.socnet.2012.01.003. 1.1, 1.5.1

[204] Wenjun Wang, Fei Cai, Pengfei Jiao, and Lin Pan. A perturbation-based framework for link prediction via non-negative matrix factorization. *Scientific Reports*, 6:38938, 2016. doi: 10.1038/srep38938. 3.2.1, 3.2.2, 3.2.3, 3.5.2, 4.1.1, 4.2.2

[205] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998. 2.2.5

[206] Nils B. Weidmann. A closer look at reporting bias in conflict event data. *American Journal of Political Science*, 60(1):206–218, 2016. doi: https://doi.org/10.1111/ajps.12196. 1.7.1, 3.2.5

[207] Halbert White. A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity. *Econometrica*, 48(4):817–838, 1980. doi: 10.2307/1912934. 2.2.4

[208] Michael Wolfowicz, Yael Litmanovitz, David Weisburd, and Badi Hasisi. A field-wide systematic review and meta-analysis of putative risk and protective factors for radicalization outcomes. *Journal of Quantitative Criminology*, 36:407–447, 2020. doi: 10.1007/s10940-019-09439-4. 1.5.3

[209] Tianyu Xia, Yijun Gu, and Dechun Yin. Research on the link prediction model of dynamic multiplex social network based on improved graph representation learning. *IEEE Access*, 9:412–420, 2021. doi: 10.1109/access.2020.3046526. 3.1.1, 6.2.3

[210] Jennifer Xu, Byron Marshall, Siddharth Kaza, and Hsinchun Chen. Analyzing and visualizing criminal network dynamics: A case study. In *Intelligence and security informatics: Second symposium on intelligence and security informatics, ISI 2004*, volume 3073

of *Lecture Notes in Computer Science*, pages 359–377. Springer-Verlag, 2004. ISBN 9783540221258. doi: 10.1007/978-3-540-25952-7_27. 1.5.2

[211] Xiaoya Xu, Bo Liu, Jianshe Wu, and Licheng Jiao. Link prediction in complex networks via matrix perturbation and decomposition. *Scientific Reports*, 7:14724, 2017. doi: 10.1038/s41598-017-14847-2. 3.2.2, 4.2.2, **??**

[212] Hansheng Xue, Luwei Yang, Wen Jiang, Yi Wei, Yi Hu, and Yu Lin. Modeling dynamic heterogeneous network for link prediction using hierarchical attention with temporal RNN. In *Machine Learning and Knowledge Discovery in Databases*, pages 282–298, 2021. doi: 10.1007/978-3-030-67658-2_17. 3.1.1, 3.5.1, 6.2.3

[213] Xiao Yang, Zhen Tian, Huayang Cui, and Zhaoxin Zhang. Link prediction on evolving network using tensor-based node similarity. In *2012 IEEE 2nd International Conference on Cloud Computing and Intelligence Systems*, pages 154–158, 2012. doi: 10.1109/CCIS.2012.6664387. 3.2.1

[214] Yang Yang, Ryan N. Lichtenwalter, and Nitesh V. Chawla. Evaluating link prediction methods. *Knowledge and Information Systems*, 45(3):751–782, 2015. doi: 10.1007/s10115-014-0789-0. 1.1, 3.1.1, 3.2.2, 3.2.3, 6.1.1

[215] Leon Yin. Smappnyu/urlexpander: Initial release, 2018. 2.2.5

[216] Rose Yu, Huida Qiu, Zhen Wen, ChingYung Lin, and Yan Liu. A survey on social media anomaly detection. *SIGKDD Explorations*, 18(1):1–14, 2016. doi: 10.1145/2980765.2980767. 6.4

[217] An Zeng and Giulio Cimini. Removing spurious interactions in complex networks. *Physical Review E*, 85(3):036101, 2012. doi: 10.1103/PhysRevE.85.036101. 1.5.1, 3.1.1

[218] Peng Zhang, An Zeng, and Ying Fan. Identifying missing and spurious connections via the bi-directional diffusion on bipartite networks. *Physics Letters A*, 378(32):2350–2354, 2014. doi: 10.1016/j.physleta.2014.06.011. 3.1.1

[219] Ting Zhang, Kun Zhang, Laishui Lv, Xun Li, and Yue Fang. Tensor decomposition for link prediction in temporal directed networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(1):013402, 2021. doi: 10.1088/1742-5468/abd310. 3.2.1, 3.2.2, 3.2.5, 3.5.2

[220] Xue Zhang, Chengli Zhao, Xiaojie Wang, and Dongyun Yi. Identifying missing and spurious interactions in directed networks. *International Journal of Distributed Sensor Networks*, 2015:27, 2015. doi: 10.1155/2015/507386. 1.5.1, 3.1.1, 3.4.2, 3.5.2

[221] Tao Zhou. Discriminating abilities of threshold-free evaluation metrics in link prediction. *Physica A: Statistical Mechanics and its Applications*, 615:128529, 2023. doi: 10.1016/j.physa.2023.128529. 3.2.2, 3.2.3

[222] Tao Zhou, Linyuan Lü, and Yi-Cheng Zhang. Predicting missing links via local information. *The European Physical Journal B*, 71(4):623–630, 2009. doi: 10.1140/epjb/e2009-00335-8. 3.2.2