Graph Walks and Graphical Models

William W. Cohen

March 2010
CMU-ML-10-102

Carnegie Mellon

Graph Walks and Graphical Models

William W. Cohen

March 2010
CMU-ML-10-102

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

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

This research is supported by grants from the National Science Foundation and the National Institutes of Health.

Keywords: graphical models, Markov random fields, personalized PageRank, random walk with restart, graph similarity

Abstract

Inference in Markov random fields, and development and evaluation of similarity measures for nodes in graphs, are both active areas of data-mining research. In this paper, we demonstrate a formal connection between inference in tree-structured Markov random fields and personalized PageRank, a widely-used similarity measure for graph nodes based on graph-walks. In particular we show a connection between computation of marginal probabilities in tree-structured discrete-variable pairwise MRFs, and computation of similarity between vertices of a graph using the personalized PageRank measure: roughly speaking, for these MRFs, computing a marginal probability $\Pr(X_i = j)$ can be reduced to computing a small set of personalized-PageRank similarity vectors, followed by a very limited postprocessing stage.

1 Introduction

Developing and evaluating useful measures for the similarity of nodes in a graph is an active area of data-mining research (e.g., [26, 4, 18, 27, 32, 24]), and one widely-used family of similarity measures are based on random walks on a graph [26, 11, 12, 3, 32]. Graph-walk based similarity measures have been used for various applications including information retrieval [3], schema matching [22], word-sense disambiguation [33], entity resolution [7], and personal information management tasks [24]. Novel techniques have been explored for computing these measures efficiently (e.g., [13, 9, 32, 6]) or tuning them to specific tasks (e.g., [23, 10, 33, 1]). These measures are closely related to similarity measures widely used for semi-supervised learning (e.g., [16, 35]) and spectral clustering (e.g., [21]) and are broadly similar to *spreading activation*, a model of human cognition that has been actively studied for nearly 40 years (e.g., [29, 2]).

Graphs are also used heavily in another active research area: in research on learning and inference with *probabilistic models*, graphical probabilistic models such as Bayes networks and Markov random fields (MRFs) are widely used (e.g., [28], [14],[5, Chapter 8]). MRFs are used in the well-known *junction-tree algorithm* for inference in Bayes networks [8]. *Conditional random fields*—i.e., MRFs tuned to optimize conditional probability of one set of variables given another—are also widely used for structured learning problems (e.g., [17, 31, 20]). MRFs are also used as an inferential "building block" in *Markov logic networks* [30], a well-studied first-order probabilistic model.

While MRFs are generally visualized as undirected graphs, they are graphs with a very special internal structure, and it is unclear to what extent data-mining techniques developed for other graphs can be usefully applied to MRFs. In this paper, we demonstrate a formal connection between graph-walk based similarity measures and inference in MRFs. More specifically, we show a connection between computation of marginal probabilities in certain MRFS, and computation of similarity between vertices of a graph using the widely-used *personalized PageRank (PPR) measure* [26], also known as *random walk with restart* [32]. Our result shows that for certain MRFs, computing a marginal probability $\Pr(X_i = j)$ can be reduced to computing a small set of PPR similarity measures, followed by a very limited postprocessing stage.

A little more precisely, our results hold for any tree-structured MRFs with discrete variables and binary cliques—a class that includes most MRFs used in conditional random fields or generated by the junction tree algorithm. We show that for an MRF of this form with L leaves and $|\mathcal{Y}|$ values of the variable X_i, then the marginal probability distribution $\Pr(X_i)$ can be approximated arbitrarily well in $O(L)$ time using $|\mathcal{Y}|$ calls to an oracle that computes a *PPR ranking vector*, where a PPR ranking vector is simply the result of a personalized-PageRank similarity computation. The constructions used in the proof are quite simple, and give an interesting insight into probabilistic inference in MRFs. Some consequences of this result are discussed in Section 4.

The remainder of the paper is organized as follows. After presenting some background material on graphs, MRFs, and similarity computation, we present in Section 3.1 a simplified version of the main result, in the form of a theorem relating MRF inference to a similarity measure for graphs that we call "all-paths similarity": this measure is not widely used in experimental practise, but is more transparently related to MRF inference. We then present two sets of corollaries of this result, which allow us to relate MRF inference to more efficient computations of all-paths similarity. In

Section 3.4 we extend the result of Section 3.1 to the more commonly-used personalized PageRank measure, and show a connection between MRF inference and computation of PPR ranking vectors over a certain family of directed graphs, and in Section 3.5 we extend this result to PPR ranking vectors over undirected graphs. Finally, in Section 3.6 we experimentally investigate certain convergence issues which are not tightly bounded by our theoretical results, and in Section 4 we conclude with a summary of the results, a discussion of related work, and a discussion of the consequences of these results.

2 Background

2.1 Graphs

Graphs arise in many contexts within computer science: for instance, collections of hypertext, social networks, and protein-protein interactions are commonly formalized as graphs. Formally, a (directed) graph $G = (V, E)$ has a set of vertices V and a set of edges $E \subseteq V \times V \times \mathcal{R}$, where an edge (v, v', w) is a directed link from v to v' with weight w. We will assume here that edge weights w lie in the range $0 \leq w < 1$. A graph is *undirected* iff each edge has an inverse with the same weight, and *acyclic* iff there are no paths from a vertex to itself.

A *path through a graph* G is a sequence of triples

$$p = \langle (v_0, v_1, w_1), (v_1, v_2, w_2), \ldots, (v_{T-1}, v_T, w_T) \rangle$$

such that every triple (v_{t-1}, v_t, w_t) is in E. The *weight* of the path p is defined as $weight(p) \equiv \prod_{t=1}^{T} w_t$. The set of all paths from v to v' is written $paths(G, v, v')$. Here and elsewhere, the argument G will be omitted when it is clear from context.

Sometimes it is convenient to think of V as the set of integers $\{1, \ldots, |V|\}$, and to think of E as a *weight matrix* $\mathbf{W}$. Formally, the weight matrix $\mathbf{W}$ for a graph $G = (V, E)$ is a $|V| \times |V|$ matrix such that $\mathbf{W}[v, v']$ is the weight of the edge from v to v' (or zero if no such edge exists). We use $\mathbf{e}_v$ to denote a unit vector with $|V|$ components, all of which are zero except for the v-th component, which is one.

2.2 "All-paths" similarity

It is often useful to define some notion of "closeness" or *similarity* over the vertices in a graph. If the edge-weights in a graph are all in the interval $[0, 1)$, then one reasonable notion of similarity for two vertices v, v' in a directed acyclic graph (DAG) G is simply the total weight of all the paths between v and v':

$$SIM_G^{AP}(v, v') \equiv \sum_{p \in paths(G,v,v')} weight(p) \tag{1}$$

According to this measure, v and v' are highly similar if they are connected by many strongly-weighted paths, and less similar if they are connected by only a few weakly-weighted paths. (Since edge-weights are strictly less than one, longer paths will necessarily have smaller weights in this measure.)

Input: A vertex v; graph $G = (V, E)$ with weight matrix $\mathbf{W}$.
Optional input: parameter $\gamma : 0 < \gamma < 1$.

Output: $APV(G, v)$: i.e., a ranking vector $\mathbf{s}$, where $SIM_G^{AP}(v, v')$ is given by $\mathbf{s}[v']$.

1. Let $\mathbf{r}_0 = \mathbf{e}_v$.
2. Let $\mathbf{s} = \mathbf{r}_0$.
3. For $t = 1, \ldots, |V|$
   (a) Let $\mathbf{r}_t = \mathbf{r}_{t-1} \cdot \mathbf{W}$
   (b) Let $\mathbf{s} = \mathbf{s} + \mathbf{r}_t$
4. Return $\mathbf{s}$.

Output: $PPV(G, v)$: i.e., a ranking vector $\mathbf{s}$, where $SIM_{G,\gamma}^{PPR}(v, v')$ is given by $\mathbf{s}[v']$.

1. Let $\mathbf{r}_0 = (1 - \gamma)\mathbf{e}_v$.
2. Let $\mathbf{s} = \mathbf{r}_0$.
3. For $t = 1, \ldots,$ to convergence:
   (a) Let $\mathbf{r}_t = \mathbf{r}_{t-1} \cdot \gamma\mathbf{W}$
   (b) Let $\mathbf{s} = \mathbf{s} + \mathbf{r}_t$
4. Return $\mathbf{s}$.

Figure 1: Computing ranking vectors for the all-paths similarity metric and the personal PageRank similarity metric

The set $paths(v, v')$ can be infinite for graphs with cycles, and can be exponentially large even for DAGS. However, there is a simple dynamic programming algorithm for finding the total weight of all paths of length t that start at p. This algorithm can be easily implemented using matrix operations: since $\mathbf{W}^2[v, v']$ is the total weight of all length-2 paths from v to v', and likewise $\mathbf{W}^t[v, v']$ is the total weight of all length-t paths from v to v', and since no paths in a DAG can be longer than $|V|$, the algorithm on the left-hand side of Figure 1 correctly computes $SIM_G^{AP}(\cdot, \cdot)$ for any DAG G.

We will call the vector $\mathbf{s}$ returned by Algorithm 1 the *all-paths ranking vector for vertex v in G*, and write it as $APV(G, v)$, or $APV(v)$ when G is clear from context. Note that $APV(v) = \sum_{t=1}^{|V|} \mathbf{e}_v \cdot \mathbf{W}^t$.

Since computing all-paths similarity to v requires computing a ranking vector for v, it is nearly as easy to compute the similarity of v to many vertices as to a single vertex. We will consider an extended version of all-paths similarity as follows: if $\mathcal{V}$ is a set of vertices, then the all-paths similarity of v to $\mathcal{V}$ is the product of the all-paths similarity of v to the elements of $\mathcal{V}$: i.e.,

$$SIM_G^{AP}(v, \mathcal{V}) \equiv \prod_{v' \in \mathcal{V}} SIM_G^{AP}(v, v')$$

2.3 Personalized PageRank

The all-paths similarity metric is closely related to another widely-used metric which we will call here *personalized PageRank similarity* [26]. An algorithm for computing a ranking vector for the personalized PageRank similarity metric is shown on the right-hand side of Figure 1. We call the output of this algorithm the *personalized PageRank ranking vector for v*, and denote it as $PPV(G, v)$, and we define $SIM_{G,\gamma}^{PPR}(v, v')$ as the analogous similarity metric between v and v',

i.e.,

$$SIM^{PPR}_{G,\gamma}(v, v') \equiv PPV(G, v)[v']$$

More formally, personalized PageRank is usually described as the result of a random walk process on a graph. Assume that G is a graph such that for every node, the sum of the weights of outgoing edges is exactly one—i.e., the sum of weights in each row of $\mathbf{W}$ is exactly one. View these weights as the probability of leaving a vertex via that edge, and now imagine a "random surfer" particle, which probabilistically traverses the graph G as follows: at each time step, with probability $1 - \gamma$, the surfer "teleports" to the "start vertex" v; and with probability $\gamma\mathbf{W}[v_1, v_2]$, the surfer moves from its current location v_1 to some new location v_2.

Now consider the point where this process converges, i.e., the probability distribution $\mathbf{p}^*_\mathbf{v}$ such that

$$\mathbf{p}^*_\mathbf{v} = (1 - \gamma)\mathbf{e}_v + \gamma\mathbf{p}^*_\mathbf{v}\mathbf{W}$$

Solving this for $\mathbf{p}^*_\mathbf{v}$ yields

$$\mathbf{p}^*_\mathbf{v} = (1 - \gamma)\mathbf{e}_v \cdot (\mathbf{I} - \gamma\mathbf{W})^{-1} \tag{2}$$

This "random surfer" process is precisely the model that underlies traditional PageRank [26], except that in traditional PageRank, the surfer "teleports" to a vertex v' chosen uniformly from V, instead of teleporting to the designated "start vertex" v.

Although Equation 2 can be computed directly, inverting the matrix $(\mathbf{I} - \gamma\mathbf{W})$ is expensive for a large graph. An alternative method for computing $\mathbf{p_v}^*$ is the "power iteration method", which relies on the fact that

$$(\mathbf{I} - \gamma\mathbf{W})^{-1} = \lim_{n\to\infty} \sum_{i=0}^{n} (\gamma\mathbf{W})^n$$

for weight matrices $\mathbf{W}$ that are normalized as described above. (A somewhat more general version of this fact is proved below, in Section 3.5.) This leads to the following proposition:

Proposition 1 *If* $\mathbf{W}$ *is normalized so that every row sum is one, then the algorithm on the right-hand side of Figure 1 converges and returns a vector* $\mathbf{s}$ *such that* $\mathbf{s} = \mathbf{p}^*_v$*, where* $\mathbf{p}^*_v$ *is the stationary distribution defined in Equation 2.*

Proof: The vector computed by the algorithm is

$$\mathbf{s} = \lim_{n\to\infty} (1 - \gamma)\mathbf{e}_v \left(\sum_{t=0}^{n} (\gamma\mathbf{W})^n \right) = (1 - \gamma)\mathbf{e}_v \lim_{n\to\infty} \left(\sum_{t=0}^{n} (\gamma\mathbf{W})^n \right) = (1 - \gamma)\mathbf{e}_v(\mathbf{I} - \gamma\mathbf{W})^{-1}$$

Personalized PageRank similarity can be viewed as a variant of all-paths similarity, obtained by downweighting paths of length n by a factor of $(1 - \gamma)\gamma^n$. The proposition above formalizes this statement: clearly, one could equivalently define $SIM^{PPR}_{\cdot,\gamma}(\cdot, \cdot)$ as

$$SIM^{PPR}_{G,\gamma}(v, v') \equiv (1 - \gamma) \sum_{p \in paths(G,v,v')} weight(p) \cdot \gamma^{|p|} \tag{3}$$

2.4 Markov random fields

A *Markov random field* $F = (\mathbf{X}, \Phi)$ is defined by a vector $\mathbf{X} = \langle X_1, \ldots, X_N \rangle$ of random variables, each of which takes one of the discrete values[1] in $\mathcal{Y} = \{1, \ldots, |\mathcal{Y}|\}$, and a set of *potential functions* Φ. Here we will consider only "pairwise" MRFs (with cliques of size two) so the potential functions are $\Phi_{i,i'} : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathcal{R}$, where $\mathcal{R}$ denotes the real numbers. We use $\Phi(X_i = j, X_{i'} = j')$ to denote $\Phi_{i,i'}(j, j')$. The set of pairs (i, i') over which Φ is defined will be denoted E_Φ.

An MRF defines a probability function over the possible assignments $\mathbf{x} = \langle x_1, \ldots, x_n \rangle$ to the variables $\mathbf{X}$ as follows:

$$\Pr(\mathbf{X} = \mathbf{x}) = \frac{1}{Z} \prod_{(i,i') \in E_\Phi} \Phi(X_i = x_i, X_{i'} = x_{i'}) \tag{4}$$

where $Z = \sum_{\mathbf{z} \in \mathcal{Y}^N} \prod_{(i,i') \in E_\Phi} \Phi(X_i = z_i, X_{i'} = z_{i'})$. An MRF of this sort is commonly visualized as a graph in which the variables are vertices and the pairs in E_Φ are (undirected) edges. For such a graph to be meaningful as an MRF, of course, it must be accompanied by additional annotations—the values assumed by the potential function. (To emphasize the difference between MRF-variable graphs, the simpler graph structure defined in Section 2.1 will sometimes be called an *ordinary graph*.) Figure 2(A) shows an MRF containing five variables, each defined over the domain $\mathcal{Y} = \{1, 2\}$, along with some of the annotations required to define the associated probability distribution over $\mathbf{X}$.

A *tree-structured MRF* is an MRF where this graph forms a tree (i.e., there is exactly one path in the graph between any pair of variables). A *leaf variable* X_k in a tree-structured MRF belongs to only one pair in E_Φ. The MRF of Figure 2(A) is tree-structured with three leaves, X_1, X_3, and X_5.

Tree-structured MRFs are important because certain types of inferences can be performed efficiently: notably, one can efficiently compute the *marginal probability* of a variable X_i taking value j, which is defined to be

$$\Pr_F(X_i = j) \equiv \frac{1}{Z} \sum_{\mathbf{x} \in \mathcal{X}_{ij}} \prod_{(i,i') \in E_\Phi} \Phi(X_i = x_i, X_{i'} = x_{i'})$$

where $\mathcal{X}_{ij} \equiv \{\mathbf{x} \in \mathcal{Y}^N : x_i = j\}$ and Z is the normalization constant defined above. It will also be useful to consider the unnormalized version of this quantity, the *belief from F for $X_i = j$*, which we will write as $Bel_F(X_i = j)$, and define as:

$$Bel_F(X_i = j) \equiv \sum_{\mathbf{x} \in \mathcal{X}_{ij}} \prod_{(i,i') \in E_\Phi} \Phi(X_i = x_i, X_{i'} = x_{i'}) \tag{5}$$

In an MRF, if all paths between variables X_i and $X_{i'}$ pass through a third variable X_s, then X_i and $X_{i'}$ are *conditionally independent* given X_s (i.e., $\Pr(X_i, X_{i'} | X_s) = \Pr(X_i | X_s) \Pr(X_{i'} | X_s)$). For instance, in the graph of the figure, X_3 is conditionally independent of X_4 given X_2.

[1] It is simple to extend any of our results to allow each X_i to have a separate domain $\mathcal{Y}_i$.

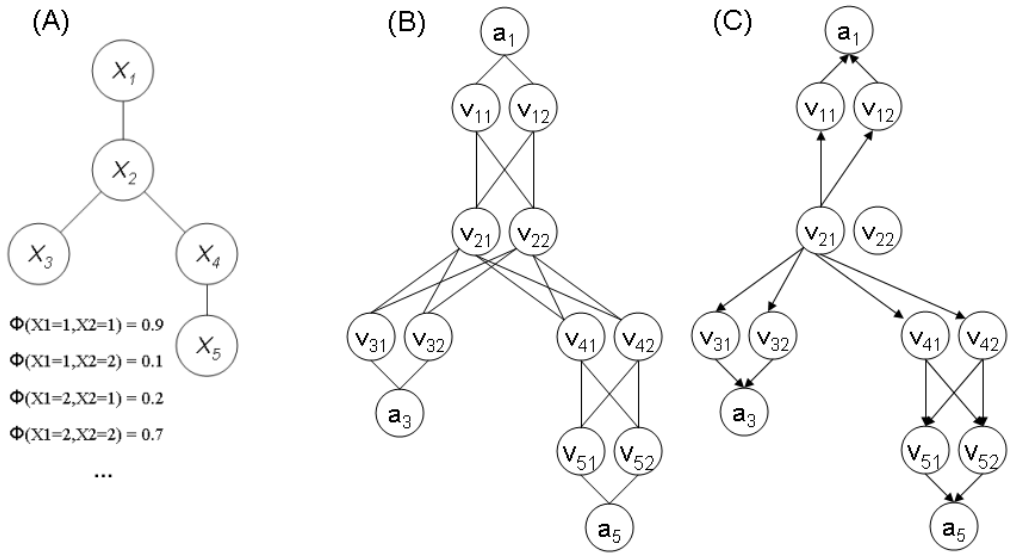

Figure 2: (A) An example MRF $F = (\langle X_1, X_2, X_3, X_4, X_5 \rangle, \Phi)$. (B) The analogous ordinary graph $\hat{G}_u = (\hat{V}, \hat{E}_u)$. (C) A directed version of the analogous graph, $\hat{G}_d(v_{2,1})$, with edges directed away from $v_{2,0}$.

3 Results

3.1 All-paths similarity and MRF inference

We will next show that all-paths similarity is in some sense "closely related" to marginal probabilities in tree-structured MRFs. By "closely related", we simply mean than all-paths similarity computations can be used to help compute marginals in a tree-structured MRF. We formalize this idea in two stages. First, we define the "ordinary graph analog" of an MRF—an ordinary graph with a similar structure. We then show that one can compute marginal probabilities in the MRF in sublinear time, given an oracle for all-paths ranking vectors on the MRF's graph analog.

More precisely, suppose that $\mathcal{G} = \{G_1, \ldots, G_i, \ldots, \}$ is a set of graphs, and that F is an MRF with $|\Phi_E|$ edges and N variables over the domain $\mathcal{Y}$, such that L variables are leaves. We say that function $g(F)$ *can be computed in* $(\mathcal{G}, APV)$*-oracle time* $O(t(L, |\Phi_E|, N, |\mathcal{Y}|))$ if $g(F)$ can be computed in the stated time, using calls to an oracle that computes, in unit time, the ranking vector $APV(G, v)$ for any graph $G \in \mathcal{G}$ and any vertex $v \in G$.

The central connection between all-paths similarity and MRF inference is given in the following lemma.

Theorem 1 *For every tree-structured MRF F, there is a family of graphs $\mathcal{G}$ (of size polynomial in L, M, N, and $|\mathcal{Y}|$) such that any marginal probability $\Pr_F(X_i = j)$ can be computed in $(\mathcal{G}, APV)$-oracle time $O(L|\mathcal{Y}|)$.*

For many interesting graphs, L will be smaller than N—for instance, in a linear-chain MRF, such as are commonly used for sequential tagging problems [17, 31, 20], there are only two "leaves", so $L = 2$. Thus the theorem above says that using an APV oracle can reduce MRF inference to *sublinear* time. In fact, $\Pr_F(X_i = j)$ can be computed with $|\mathcal{Y}|$ calls to the oracle, followed by $O(L)$ postprocessing time for each call; further, the computation performed given the all-paths oracle's result is extremely simple, consisting of only a multiplication of certain vector components and a normalization step.

The proof of the result follows. It is based a very simple and natural construction, which is illustrated by example in Figure 2, and which is defined precisely below.

Definition 1 *(Ordinary-graph analog of an MRF.) Let $F = (\mathbf{X}, \Phi)$ be a tree-structured MRF. The* ordinary-graph analog *of F is an undirected graph $\hat{G}_u^F = (\hat{V}, \hat{E}_u)$ such that*

1. *For each variable X_i in F and each possible value j for X_i, $\hat{V}$ contains a node $v_{i,j}$.*
2. *For each edge (i, i') in E_Φ and each possible $j \in \mathcal{Y}$, $j' \in \mathcal{Y}$, $\hat{E}_u$ contains an undirected edge $(v_{i,j}, v_{i',j'}, w)$, where $w = \Phi(X_i = j, X_{i'} = j')$.*
3. *For each variable leaf variable X_k in F and each possible $j \in \mathcal{Y}$, $\hat{V}$ contains a node a_k, and $\hat{E}_u$ contains an undirected edge from $(a_k, v_{k,j}, 1)$.*
4. *$\hat{V}$ and $\hat{E}_u$ contain no other nodes or edges.*

We will call the nodes a_k that are defined by step 3 the *anchor nodes* of the graph, and denote the set of anchor nodes by $\hat{A}$. Note that the size of $\hat{G}$ is linear in the size of F: if F has N variables, of which L are leaves, and Φ has M edges, then $\hat{G}$ has $(N|\mathcal{Y}| + L)$ vertices and $(M|\mathcal{Y}|^2 + 2L)$ edges. (Note that Φ may also have up to $|\mathcal{Y}|^2$ parameters for each edge.)

Figure 2(B) shows the ordinary-graph analog of the MRF of Figure 2(A). For this graph, $\hat{A} = \{a_1, a_3, a_5\}$.

We are actually most interested in certain set $\mathcal{G}$ of *directed* graphs that are derived from the analog of F, as follows.

Definition 2 *(Graph analog directed away from v.) If $\hat{G}_u^F = (\hat{V}, \hat{E}_u)$ is the ordinary-graph analog of the MRF F, let $\hat{G}_{d(v)}^F = (\hat{V}, \hat{E}_{d(v)})$ be the directed graph that is obtained by directing each edge "away from v". More precisely, if $v_{i,j}$ is in $\hat{G}_u$, then let us call the vertices $v_{i,j'}$ the* alternatives to $v_{i,j}$*, and define $dist(v, v')$, the "distance" from v to v', as follows*

$$dist(v, v') \equiv \begin{cases} 0 \text{ if } v' \text{ is an alternative to } v \\ \min_{p \in paths(\hat{G}_u^F, v, v')} |p| \text{ otherwise} \end{cases}$$

Now let $\hat{E}_{d(v)}$ contain the edges $(v_1, v_2, w) \in \hat{E}_u$ such that $dist(v, v_1) < dist(v, v_2)$. The directed graph $\hat{G}_{d(v)}^F = (\hat{V}, \hat{E}_{d(v)})$ is called the ordinary-graph analog of the MRF F directed away from v.

Figure 2(C) shows the ordinary-graph analog of the MRF of Figure 2(A), directed away from $v_{2,1}$.

The following lemma presents the main intuition behind the result of Theorem 1.

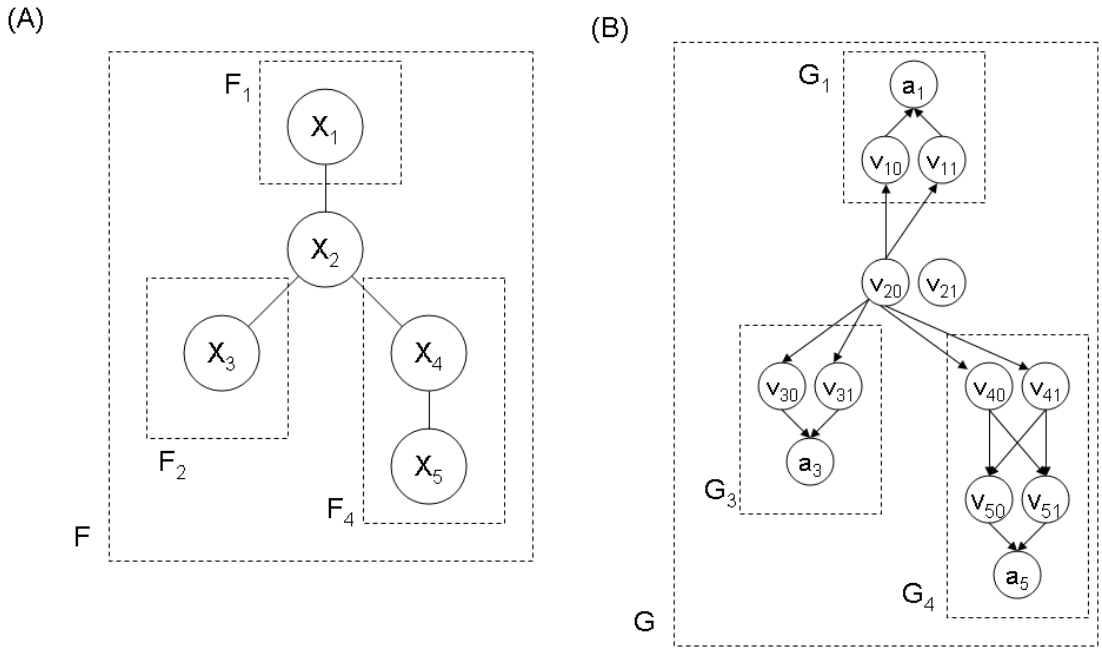

Figure 3: (A) The inductive step for Lemma 1, illustrating the blanket of an example MRF. (B) The inductive step for Lemma 1, illustrating the paths from $v_{2,0}$.

Lemma 1 *If F is a MRF and $\hat{G}^F_{d(v_{i,j})}$ is the ordinary-graph analog of F directed away from $v_{i,j}$, then the belief from F for $X_i = j$ is equivalent to the all-paths similarity of $v_{i,j}$ to the set of anchor nodes $\hat{\mathcal{A}}$ in $\hat{G}^F_{d(v_{i,j})}$. In other words,*

$$Bel_F(X_i = j) = SIM^{AP}_{\hat{G}^F_{d(v_{i,j})}}(v_{i,j}, \hat{\mathcal{A}})$$

Proof: By induction on N, the number of MRF variables. For brevity, we will use $\hat{G}$ for $\hat{G}_{d(v_{i,j})}$ below.

When the MRF contains a single variable X_1, then $Bel(X_1 = j) = 1$ for all j. (With no edges, Equation 5 vacuously assigns the value of 1 to the product $\prod_{(i,i')\in E_\Phi} \Phi(X_i = x_i, X_{i'} = x_{i'})$.) Likewise, v_{1j} is connected by a unit-weight link directly to the single vertex $a \in \hat{\mathcal{A}}$, so $SIM^{AP}_{\hat{G}}(v_{i,j}, \hat{\mathcal{A}}) = 1$.

For the inductive step, assume the lemma holds for MRFs of fewer than N variables, and consider the the variables $X_{m_1}, \ldots, X_{m_B}$ that are directly connected to X_i by an edge in E_ϕ. (These variables are called *Markov blanket* of X_i.) Imagine that we removed X_i from F: then since F is tree-structured, the subparts of F that contain $X_{m_1}, \ldots, X_{m_B}$ respectively would be disconnected from each other, and would hence comprise independent MRFs $F_{m_1}, \ldots, F_{m_B}$.

Figure 3(A) illustrates this construction for the MRF of Figure 2(A) and the variable X_2. The Markov blanket of X_2 are the variables X_1, X_3, X_4 (i.e., $B = 3$ and $m_1 = 1$, $m_2 = 3$, and $m_3 = 4$.) Removing X_2 from F produces the three smaller MRFs identified as F_1, F_3, F_4 in the figure.

Using the independence of the X_{m_b}'s given X_i, we have

$$Bel_F(X_i = j) = \prod_{b=1}^{B} \sum_{j' \in \mathcal{Y}} Bel_{F_{m_b}}(X_{m_b} = j') \Phi(X_i = j, X_{m_b} = j') \tag{6}$$

For those familiar with MRF's, Equation 6 can be established from the independence of the X_{m_b}'s given X_i. (However, Equation 6 also follows quite directly from Equation 4. A short proof of this is given in Appendix A.)

We now turn to $SIM^{AP}_{\hat{G}}(v_{i,j}, \hat{\mathcal{A}})$. Imagine that we removed v_{ij} from $\hat{G}$: since F is tree-structured, this would split $\hat{G}$ into disconnected subgraphs, which we will denote $\hat{G}_{m_1}, \ldots, \hat{G}_{m_B}$ respectively. We will also use $v_{m_b,1}, \ldots, v_{m_b,|\mathcal{Y}|}$ to denote the roots of $\hat{G}_{m_b}$ (i.e., the vertices that were connected to $v_{i,j}$ in $\hat{G}$.)

Figure 3(B) illustrates this construction. The graph shown in the figure is $\hat{G}_{d(v_{2,1})}$. The disconnected subgraphs are labeled G_1, G_3, G_4.

Let $\hat{\mathcal{A}}_{m_b}$ be the subset of $\hat{\mathcal{A}}$ contained in $\hat{G}_{m_b}$. (For instance $\hat{\mathcal{A}}_4$ in Figure 3(B) would be the singleton set $\{a_5\}$.) Clearly all paths to a vertex in $\hat{\mathcal{A}}$ must pass through one of the roots $v_{m_b,j'}$, so

$$SIM^{AP}_{\hat{G}}(v_{ij}, \hat{\mathcal{A}}) = \prod_{b=1}^{B} \sum_{j' \in \mathcal{Y}} SIM^{AP}_{\hat{G}_{m_b}}(v_{m_b,j'}, \hat{\mathcal{A}}_{m_b}) \cdot \mathbf{W}[v_{i,j}, v_{m_b,j'}]$$

By construction $\mathbf{W}[v_{i,j}, v_{m_b,j'}] = \Phi(X_i = j, X_{m_b} = j')$. Also, $\hat{G}_{m_b}$ is almost identical[2] to $\hat{G}^{F_{m_b}}_{d(v_{m_b,j'})}$, the ordinary-graph analog of F_{m_b} directed away from $v_{m_b,j'}$, and it is easily verified that the differences between $\hat{G}_{m_b}$ and $\hat{G}^{F_{m_b}}_{d(v_{m_b,j'})}$ do not affect computation of all-paths similarity. Thus

$$SIM^{AP}_{\hat{G}}(v_{ij}, \hat{\mathcal{A}}) = \prod_{b=1}^{B} \sum_{j' \in \mathcal{Y}} SIM^{AP}_{\hat{G}^{F_{m_b}}_{d(v_{m_b,j'})}}(v_{m_b,j'}, \hat{\mathcal{A}}_{m_b}) \cdot \Phi(X_i = j, X_{m_b} = j') \tag{7}$$

Now, the lemma can be proved by induction as follows. First rewrite Equation 6, using the inductive hypothesis to replace $Bel_{F_{m_b}}(X_{m_b} = j')$ with $SIM^{AP}_{\hat{G}^{F_{m_b}}_{d(v_{m_b,j'})}}(v_{m_b,j'}, \hat{\mathcal{A}}_{m_b})$:

$$Bel_F(X_i = j) = \prod_{b=1}^{B} \sum_{j' \in \mathcal{Y}} SIM^{AP}_{\hat{G}^{F_{m_b}}_{d(v_{m_b,j'})}}(v_{m_b,j'}, \hat{\mathcal{A}}_{m_b})(X_{m_b} = j') \Phi(X_i = j, X_{m_b} = j') \tag{8}$$

Since the right-hand side of Equation 8 is the same as the right-hand side of Equation 7, it must be that $Bel_F(X_i = j) = SIM^{AP}_{\hat{G}}(v_{ij}, \hat{\mathcal{A}})$, concluding the proof of the lemma.

The proof of Theorem 1 is now immediate.

Proof:(of Theorem 1). Given a tree-structured MRF F, let $\mathcal{G}$ be the set of all graphs of the form $\hat{G}^F_{d(v_{i,j})}$. There are $N|\mathcal{Y}|$ such graphs, each of which has $N|\mathcal{Y}| + L$ nodes and $|E_\Phi||\mathcal{Y}|^2 + 2L$ edges, and any marginal probability $\Pr_F(X_i = j)$ can be computed as follows:

[2](The only differences are the addition in $\hat{G}_{m_b}$ of some extra edges leading away from the nodes $v_{m_b,\tilde{j}}$, for $\tilde{j} \neq j'$, and the omission in in $\hat{G}^F_{m_b}$ of anchor node a_{m_b} for the leaf variable X_{m_b}, which is the root of G_{m_b}.)

Input: A vertex v; graph $G = (V, E)$ with weight matrix $\mathbf{W}$.
Optional input: parameter $\gamma : 0 < \gamma < 1$.

Output: $APV^{dir}(G, v) \equiv APV(G_{d(v)}, v)$

- $*$Let $\mathbf{d}_0 = \sum_{v' \in S} \mathbf{e}_{v'}$, where S contains the alternatives to v, plus v itself.
- Let $\mathbf{r}_0 = \mathbf{e}_v$.
- Let $\mathbf{s} = \mathbf{r}_0$.
- For $t = 1, \ldots, |V|$
  - Let $\mathbf{r}_t = \mathbf{r}_{t-1} \cdot \mathbf{W}$
  - $*$For all $v' : \mathbf{r}_t[v'] \neq 0$,
    - $*$Let $\mathbf{d}[v'] = \min(\mathbf{d}[v'], t + 1)$
    - $*$If $\mathbf{d}[v'] < t + 1$ then let $\mathbf{r}_t[v'] = 0$
  - Let $\mathbf{s} = \mathbf{s} + \mathbf{r}_t$
- Return $\mathbf{s}$.

Output: $PPV^{dir}(G, v) \equiv PPV(G_{d(v)}, v)$

- $*$Let $\mathbf{d}_0 = \sum_{v' \in S} \mathbf{e}_{v'}$, where S contains the alternatives to v, plus v itself.
- Let $\mathbf{r}_0 = \mathbf{e}_v$.
- Let $\mathbf{s} = (1 - \gamma)\mathbf{r}_0$.
- For $t = 1, \ldots,$ to convergence:
  - Let $\mathbf{r}_t = \mathbf{r}_{t-1} \cdot \gamma\mathbf{W}$
  - $*$For all $v' : \mathbf{r}_t[v'] \neq 0$,
    - $*$Let $\mathbf{d}_t[v'] = \min(\mathbf{d}_{t-1}[v'], t + 1)$
    - $*$If $\mathbf{d}_t[v'] < t + 1$ then let $\mathbf{r}_t[v'] = 0$
  - Let $\mathbf{s} = \mathbf{s} + \mathbf{r}_t$
- Return $\mathbf{s}$.

Figure 4: Computing the "directed" variants of the all-paths similarity metric and the personal PageRank similarity metric.

1. For each $j' \in \mathcal{Y}$:
   (a) Compute $\mathbf{s}_{i,j'} = APV(\hat{G}_{d(v_{i,j'})}, v_{i,j'})$
   (b) Using the lemma, compute[3] $Bel_F(X_i = j') = \prod_{a \in \hat{\mathcal{A}}} \mathbf{s}_{i,j'}[a]$.
2. Return $\Pr_F(X_i = j) = \frac{Bel_F(X_i=j)}{\sum_{j'} Bel_F(X_i=j')}$

This computation requires $|\mathcal{Y}|$ calls to $APV(\cdot, \cdot)$ followed by $O(L)$ time to post-process these vectors.

3.2 "Directed" APV and MRF inference

Theorem 1 shows that that reducing MRF inference to APV computation is possible. As stated, however, the theorem suggests that the reduction requires constructing many different variant graphs. In fact, this is not necessary: whenever the APV oracle is called by the algorithm of Theorem 1, it is called with arguments $APV(\hat{G}_{d(v)}, v)$ for some vertex v, and it is quite simple to compute the all-paths similarity ranking vector for v while simultaneously determining which edges from $\hat{G}_u$ belong in the directed version $\hat{G}_{d(v)}$.

[3]Here $\hat{\mathcal{A}}$ are the anchors in $\hat{G}_{d(v_{i,j'})}$.

Input: An MRF F, a variable X_i, and a value $j \in \mathcal{Y}$.
Optional input: parameter $\gamma : 0 < \gamma < 1$.
Output: An approximation to the marginal probability $\Pr_F(X_i = j)$.

- Let $\hat{G}_u$ be the ordinary-graph analog of F.
- For each $j' \in \mathcal{Y}$:
  - Perform one of the following variant steps:
    V1: Compute $\mathbf{s}_{i,j'} = APV(\hat{G}_{d(v_{i,j'})}, v_{i,j'})$
    V2: Compute $\mathbf{s}_{i,j'} = APV^{dir}(\hat{G}_u, v_{i,j'})$
    V3: Compute $\mathbf{s}_{i,j'} = PPV(\hat{G}_{d(v_{i,j'})}, v_{i,j'})$
    V4: Compute $\mathbf{s}_{i,j'} = PPV^{dir}(\hat{G}_u, v_{i,j'})$
    V5: Compute $\mathbf{s}_{i,j'} = PPV(\hat{G}_u, v_{i,j'})$
  - Compute $\tilde{B}_F(X_i = j') = \prod_{a \in \hat{\mathcal{A}}} \mathbf{s}_{i,j'}[a]$.
- Return $\frac{\tilde{B}_F(X_i=j)}{\sum_{j'} \tilde{B}_F(X_i=j')}$ as the value of $\Pr_F(X_i = j)$

Figure 5: Variants of the algorithm defined in Theorem 1.

An algorithm for doing this computation is shown on the left-hand side of Figure 4. (The starred lines correspond to additions to the algorithm of Figure 1.) The algorithm is based on the observation that while computing $\mathbf{r}_t$ (the all-paths ranking vector restricted to paths of length t or less) it is also possible to compute (an approximation to) the minimum distance from v to every node v': in Figure 4, $\mathbf{d}_t[v'] = 0$ if $dist(v, v') > t$, and $\mathbf{d}_t[v'] = dist(v, v') + 1$ otherwise, and hence it can be determined at iteration t whether or not to include an edge (a weight from $\mathbf{W}$) in the computation. We will call this variant of the APV algorithm *directed* APV. Figure 4 also shows the directed variant of the personal PageRank similarity computation.

Below we will discuss not only the effects of replacing the APV oracle for a directed graph with an APV^{dir} oracle for an undirected graph, but also the effects of replacing the APV oracle with a various types of PPV oracles. To facilitate this discussion, Figure 5 shows five variants of the algorithm used in the proof of Theorem 1. In each variant, a different ranking vector is used to compute the "belief" in a particular variable assignment. In the figure we will write this "belief" as $\tilde{B}_F(X_i = j)$—as we will see, $\tilde{B}_F(X_i = j)$ need not coincide with $B_F(X_i = j)$. Since it is straightforward to verify that $APV^{dir}(\hat{G}_u, v_{ij})$ computes the same ranking vector as $APV(\hat{G}_d(v_{i,j}), v_{ij})$, it follows that:

Proposition 2 *Variant V2 of Figure 5 is correct—i.e., it returns* $\Pr_F(X_i = j)$.

Thus, for every tree-structured MRF F*, any marginal probability* $\Pr_F(X_i = j)$ *can be computed in* $(\hat{G}_u, APV^{dir})$*-oracle time*[4] $O(L|\mathcal{Y}|)$*, where* $\hat{G}_u$ *is the ordinary-graph analog of* F.

[4]Here we use (G, APV^{dir}) as an abbreviation of $(\mathcal{G}, APV^{dir})$ for $\mathcal{G} = \{G\}$, and we define $(\mathcal{G}, APV^{dir})$-oracle

Input:

- An MRF F.
- A set S of variables X_i, corresponding to marginal-probability computations.

Preprocessing stage:

1. Create $\hat{G}_u$, the ordinary-graph analog of F.
2. For each anchor node a_k in $\hat{\mathcal{A}}$, let $\mathbf{s}_k = APV^{dir}(\hat{G}, a_k)$

   (In computing APV^{dir}, the set of "alternatives" to an anchor node a_k is defined to be the empty set.)

Computation stage:

1. For each $X_i \in S$ and each $j \in \mathcal{Y}$
   (a) Compute $Bel_F(X_i = j) = \prod_{k=1}^{L} \mathbf{s}_k[v_{i,j}]$
   (b) Compute $\Pr_F(X_i = j) = Bel_F(X_i = j) / \sum_{j'} Bel_F(X_i = j')$.

Figure 6: Computing many marginal probabilities efficiently with an APV^{dir} oracle.

3.3 Propogating similarity from leaves to internal nodes

The algorithm of Theorem 1 is also inefficient if one needs to compute many marginal probabilities on the same graph. MRF inference is usually performed with "message-passing", where "messages" originate in the leaves of the tree and propogate inward, to be combined at internal nodes of the tree. The algorithm of Figure 6 implements a procedure with this flavor. First, the algorithm computes the all-paths ranking vectors for each anchor node $a_k \in \hat{G}$. (Recall that internally, this computation requires a traversal of $\hat{G}$ to compute the all-paths similarity of $\hat{a_k}$ to each node $v \in \hat{G}$, a process similar to propogating messages inward from a_k.) After computing the L ranking vectors, the belief $Bel_F(X_i = j)$ is be computed as

$$Bel_F(X_i = j) = \prod_{k=1}^{L} APV^{dir}(\hat{G}_u, a_k)[v_{i,j}]$$

which is *not* identical to the computation that is justified by Lemma 1, namely

$$Bel_F(X_i = j) = SIM^{AP}_{\hat{G}_{d(v_{i,j})}}(v_{i,j}, \hat{\mathcal{A}}) = \prod_{k=1}^{L} APV^{dir}(\hat{G}_u, v_{ij})[a_k]$$

time analogously to the $(\mathcal{G}, APV)$-oracle time.

However, it is easy to see that for any v_{ij} and a_k

$$APV^{dir}(\hat{G}_u, a_k)[v_{i_j}] = APV^{dir}(\hat{G}_u, v_{i,j})[a_k]$$

by simply recalling that $APV(v_1)[v_2]$ depends only on the weight of the paths between v_1 and v_2, and noting that every path in $\hat{G}_{d(v_{ij})}$ that leads to a_k can be inverted to form a path in $\hat{G}_{d(a_k)}$ that leads to v_{ij}, and vice-versa. Thus the following proposition holds:

Proposition 3 *The algorithm of Figure 6 correctly computes the marginal probabilities of each variable X_i in S. Hence*

- *For every tree-structured MRF F, any marginal probability $\Pr_F(X_i = j)$ can be computed in $(\hat{G}_u, APV^{dir})$-oracle time $O(L + |\mathcal{Y}||S|)$, where $\hat{G}_u$ is the ordinary-graph analog of F.*
- *For every tree-structured MRF F, there is a family of polynomial-sized graphs $\mathcal{G}$ such that any set of S marginal probabilities can be computed in $(\mathcal{G}, APV)$-oracle time $O(L+|\mathcal{Y}||S|)$.*

The first claim of the proposition follows directly from the correctness of the algorithm of Figure 6. The second follows from the observation that the calls to $APV^{dir}(\hat{G}_u, v)$ can be replaced with $APV(\hat{G}_{d(v)}, v)$.

3.4 "Directed" personalized PageRank and MRF inference

Returning to the algorithms of Figure 5: from Theorem 1 we know that Variant V1 is correct, and from Proposition 2 we know that Variant V2 is correct. Let us now consider the result of using the personalized PageRank ranking in place of the all-paths similarity ranking. We have the following result.

Theorem 2 *Variants 3 and 4 of the algorithm of Figure 5 return $\Pr_F(X_i = j)$.*

Proof: We will first analyze Variant V3. For brevity, we will use $\hat{G}$ for $\hat{G}_{d(v_{i,j})}$. Note that PPV must converge in these cases, since $\hat{G}$ is a DAG. (To see this, notice that $\mathbf{W}^t[v, v']$ is the weight of all length-t paths from v to v'. Since paths in $\hat{G}_{d(v)}$ must have length no more than $|V|$, $\mathbf{W}^t = \mathbf{0}$ for $t > |V|$.)

Consider an internal node $v_{ij} \in \hat{G}$ and an anchor vertex a_k, and let us examine the relationship between $SIM^{AP}_{\hat{G}}(v_{i,j}, a_k)$ and $SIM^{PPR}_{\hat{G},\gamma}(v_{i,j}, a_k)$. All the paths in $\hat{G}$ between v_{ij} and a_k are of some fixed length, which is simply the number of variables in F between X_i and the leaf X_k that corresponds to a_k. If we write this distance as d_{ik}, so it must be that

$$SIM^{PPR}_{\hat{G},\gamma}(v_{i,j}, a_k) = (1-\gamma)\gamma^{d_{ik}} SIM^{AP}_{\hat{G}}(v_{i,j}, a_k)$$

$*$	$(1-\gamma)\gamma^2$	$\times$	$weight(v_{11} \to v_{21} \to a_2)$
$*$	$(1-\gamma)\gamma^2$	$\times$	$weight(v_{11} \to v_{22} \to a_2)$
	$(1-\gamma)\gamma^4$	$\times$	$weight(v_{11} \to v_{21} \to v_{11} \to v_{21} \to a_2)$
	$(1-\gamma)\gamma^4$	$\times$	$weight(v_{11} \to v_{21} \to v_{11} \to v_{22} \to a_2)$
!	$(1-\gamma)\gamma^4$	$\times$	$weight(v_{11} \to v_{21} \to v_{12} \to v_{21} \to a_2)$
!	$(1-\gamma)\gamma^4$	$\times$	$weight(v_{11} \to v_{22} \to v_{12} \to v_{22} \to a_2)$
	$(1-\gamma)\gamma^4$	$\times$	$weight(v_{11} \to a_1 \to v_{11} \to v_{21} \to a_2)$
	$(1-\gamma)\gamma^4$	$\times$	$weight(v_{11} \to a_1 \to v_{11} \to v_{22} \to a_2)$
!	$(1-\gamma)\gamma^4$	$\times$	$weight(v_{11} \to a_1 \to v_{12} \to v_{21} \to a_2)$
!	$(1-\gamma)\gamma^4$	$\times$	$weight(v_{11} \to a_1 \to v_{12} \to v_{22} \to a_2)$
		$\ldots$	

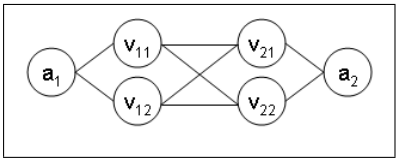

Figure 7: Behavior of $PPR(\hat{G}_u, v_{ij})$ on a sample graph

So for Variant V3, we have that

$$\begin{aligned}
\tilde{B}_F(X_i = j) &= \prod_{a_k \in \hat{\mathcal{A}}} SIM^{PPR}_{\hat{G}_{d(v_{i,j})},\gamma}(v_{i,j}, a_k) \\
&= \prod_{a_k \in \hat{\mathcal{A}}} SIM^{AP}_{\hat{G}_{d(v_{i,j})}}(v_{i,j}, a_k)(1-\gamma)\gamma^{d_{ik}} \\
&= \prod_{a_k \in \hat{\mathcal{A}}} SIM^{AP}_{\hat{G}_{d(v_{i,j})}}(v_{i,j}, a_k) \prod_{a_k \in \hat{\mathcal{A}}} (1-\gamma)\gamma^{d_{ik}} \\
&= B_F(X_i = j) \cdot c_i
\end{aligned}$$

where we define $c_i \equiv \prod_{a_k \in \hat{\mathcal{A}}}(1-\gamma)\gamma^{d_{ik}}$. Note that this distance d_{ik} is the same for all the alternatives to v_{ij}, so that this argument also holds for other values of $\tilde{B}_F(X_i = j')$. Hence, the value returned by this variant is

$$\frac{\tilde{B}_F(X_i = j)}{\sum_{j'} \tilde{B}_F(X_i = j')} = \frac{B_F(X_i = j) \cdot c_i}{\sum_{j'} B_F(X_i = j') \cdot c_i} = \Pr(X_i = j)$$

and so the variant is correct.

The correctness of Variant V4 follows from the correctness of Variant V3 and the arguments used to support Proposition 2.

To summarize the proof informally: the personalized PageRank similarity metric differs from the all-paths similarity metric only in the way it treats longer paths—specifically it downweights paths of length d by a factor of $(1-\gamma)\gamma^d$. However, while this downweighting changes the unnormalized "belief" in a variable assignment, it does not change the normalized *probability* in a variable assignment, because the downweighting equally affects the paths to a variable-assignment node v_{ij} and the paths to its alternatives.

3.5 Computing marginals with "undirected" personalized PageRank

Let us now consider Variant V5. Again, the first issue to consider is convergence, which in this case is not immediate, as $\hat{G}_u$ is not a DAG, nor need it be appropriately normalized. However, it is simple to show that:

Theorem 3 *For every MRF F, there is an MRF F' that defines an equivalent probability distribution such that the algorithm given in Figure 1 will always converge when given as its graph input $\hat{G}_u^{F'}$, the ordinary-graph analog of F'.*

Proof: We begin by demonstrating that if $\mathbf{A}$ is a matrix such that $\lim_{n\to\infty} \mathbf{A}^n = \mathbf{0}$, then

$$\lim_{n\to\infty} \sum_{t=1}^{n} \mathbf{A}^n = (\mathbf{I} - \mathbf{A})^{-1} \tag{9}$$

To see that this is true, define $\mathbf{X}_n \equiv \sum_{t=0}^{n} (\mathbf{A})^t$. Multiplying both sides of this definition by $(\mathbf{I}-\mathbf{A})$ will generate "telescoping" sums that can be simplified, as follows:

$$\begin{aligned}
\mathbf{X}_n(\mathbf{I} - \mathbf{A}) &= \left(\sum_{t=0}^{n} \mathbf{A}^t\right)(\mathbf{I} - \mathbf{A}) \\
&= (\mathbf{I} - \mathbf{A}) + (\mathbf{A} - \mathbf{A}^2) + \ldots (\mathbf{A}^n - \mathbf{A}^{n+1}) \\
&= (\mathbf{I} - \mathbf{A}^{n+1}) \\
\mathbf{X}_n &= (\mathbf{I} - \mathbf{A}^{n+1})(\mathbf{I} - \mathbf{A})^{-1}
\end{aligned}$$

Hence $\mathbf{X}_n = (\mathbf{I} - \mathbf{A}^{n+1})(\mathbf{I} - \mathbf{A})^{-1}$ and Equation 9 follows.

Hence the convergence of the PPV method requires only that $\lim_{n\to\infty}(\gamma\mathbf{W})^n = \mathbf{0}$. For this condition to hold it is sufficient that each row in $\mathbf{W}$ sum to *at most* one, rather than exactly one. This condition is easy to meet by simply replacing $\mathbf{W}$ with $\frac{1}{c}\mathbf{W}$, where $c \equiv \max_{v\in V} \sum_{v'\in V} \mathbf{W}[v, v']$; or equivalently, by dividing each potential $\Phi(X_i = j, X_{i'} = j')$ in F by c. Notice that the joint probability defined by an MRF does not change if the potential functions Φ are uniformly scaled up or down by a constant.

We will henceforth assume that the potential functions for MRF's are appropriately scaled, so that the PPV algorithm converges.

Although the algorithm converges, it is quite easy to show that this variant is *not* correct, at least in the sense defined so far. Consider the following small graph $\hat{G}$ of Figure 7 (derived from an MRF F with two variables, each with two possible values), and consider using Variant V5 to compute the "belief" for $X_1 = 1$, corresponding to the node v_{11}. For this undirected graph $\hat{G}$, $SIM^{PPR}_{\hat{G},\gamma}(v_{11}, a_2)$ will be the sum of weights of the paths shown in the figure, as well as infinitely many other paths.[5] The two paths marked with an asterisk $(*)$ are the paths that will be included in computation of $SIM^{PPR}_{\hat{G}_{d(v_{11})},\gamma}(v_{11}, a_2)$. The other paths contain loops, and are not included in

[5]The figure shows all paths of length four or less.

the directed graph. The paths marked with an exclamation point (!) are especially worrisome, as they are the paths associated with the belief for $X_1 = 2$. So it appears that rather than returning a downweighted version of the $Bel(X_1 = 1)$, the function $SIM^{PPR}_{\hat{G},\gamma}(v_{11}, a_2)$ is actually returning a weighted average of $Bel(X_1 = 1)$ and $Bel(X_1 = 2)$, as well as certain other terms associated with other sorts of paths not counted in $\hat{G}_d(v_{11})$.

However, the weight of these "undesirable" paths is substantially lower than the weight of the "correct" paths. This suggests that Variant V5 may nonetheless return a reasonably accurate approximation of $\Pr_F(X_i = j)$: in particular if γ is small then the longer paths will have only a small impact on the final result. This intuition is correct, as shown below:

Theorem 4 *Let $p_\gamma(F, X_i, j)$ be the value returned by Variant V5 of the algorithm of Figure 5. Then*

$$\lim_{\gamma \to 0} p_\gamma(F, X_i, j) = \Pr_F(X_i = j)$$

Proof: Recall from Equation 3 that

$$SIM^{PPR}_{\hat{G}_u,\gamma}(v, v') \equiv (1-\gamma) \sum_{p \in paths(\hat{G}_u, v, v')} weight(p) \cdot \gamma^{|p|} \tag{10}$$

The set $paths(\hat{G}_u, v, v')$ can be broken down into the disjoint sets $paths(\hat{G}_{d(v)}, v, v')$ and S_{loop}, where $S_{loop} \equiv paths(\hat{G}_u, v, v') - paths(\hat{G}_{d(v)}, v, v')$. If $v = v_{ij}$ and $v' = a_k$ then all the paths in $paths(\hat{G}_{d(v)}, v, v')$ are of length d_{ik}; clearly all the paths in S_{loop} are of length at least $d_{ik} + 2$. Hence we can rewrite Equation 10 as

$$SIM^{PPR}_{\hat{G}_u,\gamma}(v_{ij}, a_k) \equiv (1-\gamma)\gamma^{d_{ik}} \left(SIM^{AP}_{\hat{G}_{d(v_{ij})}}(v_{ij}, a_k) + \gamma^2 \sum_{p \in S_{loop}} weight(p) \cdot \gamma^{|p|-d_{ik}-2} \right)$$

For brevity, let $B_{ijk} = SIM^{AP}_{\hat{G}_{d(v_{ij})}}(v_{ij}, a_k)$ and let $E_{ijk} = \sum_{p \in S_{loop}} weight(p) \cdot \gamma^{|p|-d_{ik}-2}$. Note that E_{ijk} is bounded by some constant since $PPV_{\hat{G}_u}$ converges. We can now write the "belief" computed by Variant V5 as

$$\begin{aligned}
\tilde{B}(X_i = j) &= \prod_{a_k \in \hat{\mathcal{A}}} (1-\gamma)\gamma^{d_{ik}}(B_{ijk} + \gamma^2 E_{ijk}) \\
&= \left(\prod_{a_k \in \hat{\mathcal{A}}} (1-\gamma)\gamma^{d_{ik}} \right) \left((\prod_{a_k \in \hat{\mathcal{A}}} B_{ijk}) + \gamma^2 (\sum \text{high-order terms}) \right) \\
&= c_i \left((\prod_{a_k \in \hat{\mathcal{A}}} B_{ijk}) + \gamma^2 (\sum \text{high-order terms}) \right)
\end{aligned}$$

Parameters: d_{max}, the maximum depth of the tree; p_{leaf}, the probability that a node will be a leaf; $p_{ch}(k)$, the probability that a non-leaf node will have exactly k children; $p_\Phi(\phi)$, the probability that the potential between two nodes will have value $\Phi(X_i = j, X_{i'} = j;) = \phi$; and $p_J(j)$, the probability a node will have exactly J values.
To build a tree: repeatedly extend a tree at node X_i, as follows (where X_i is a variable with legal values $1, \ldots, J_i$):

1. If the depth of X_i is more than d_{max}, make X_i a leaf. Otherwise, with probability p_{leaf}, make X_i a leaf.

2. If X_i is not a leaf: (a) Pick C, the number of children of X_i, according to $p_{ch}(.)$; (b) generate new nodes $X_{c_1}, \ldots, X_{c_C}$ to be the children of X_i; (c) and for each child node X_{c_ℓ} of X_i: (i) Pick J_ℓ, the number of children of X_{c_ℓ}, according to $p_{ch}(.)$, (ii) For each $j \in \{1, \ldots, J_i\}$, $j' \in \{1, \ldots, J_\ell\}$ pick $\Phi(X_i = j, X_\ell = j')$ according to $p_\Phi(.)$.

Figure 8: Generating random tree-structured MRFs

where the "high-order terms" are various products of B_{ijk}'s and $E_{ijk'}$'s. The sum of these will henceforth be written T_{ik}, so the value returned by this variant can be written

$$\begin{aligned} p_\gamma(F, X_i, j) &= \frac{\tilde{B}_F(X_i = j)}{\sum_{j'} \tilde{B}_F(X_i = j')} \\ &= \frac{c_i \left((\prod_{a_k \in \hat{\mathcal{A}}} B_{ijk}) + \gamma^2 T_{ik} \right)}{\sum_{j' \in \mathcal{Y}} c_i \left((\prod_{a_k \in \hat{\mathcal{A}}} B_{ij'k}) + \gamma^2 T_{ik} \right)} \\ &= \frac{\left((\prod_{a_k \in \hat{\mathcal{A}}} B_{ijk}) + \gamma^2 T_{ik} \right)}{\sum_{j' \in \mathcal{Y}} \left((\prod_{a_k \in \hat{\mathcal{A}}} B_{ij'k}) + \gamma^2 T_{ik} \right)} \\ &= \frac{(Bel_F(X_i = j) + \gamma^2 T_{ik})}{\sum_{j' \in \mathcal{Y}} (Bel_F(X_i = j) + \gamma^2 T_{ik})} \end{aligned}$$

and hence $\lim_{\gamma \to 0} p_\gamma(F, X_i, j) = \Pr_F(X_i = j)$.

3.6 Experimental confirmation

The focus of this paper is on formal, not experimental results. However, while Theorem 4 shows that a "vanilla" version of personalized PageRank can be used to perform approximate inference in tree-structured MRFs, and also suggests that the approximation will be better for smaller γ, the theorem does not give any precise bounds on the quality of the approximation. To explore this issue, we conducted some experiments with Variant V5 of the algorithm of Figure 5.

We constructed a random tree-structured MRF, following the procedure described in Figure 8. The resulting ordinary-graph analog $\hat{G}_u$ contained 968 nodes and 3864 edges, with a diameter of 16. We then picked 100 variable-value pairs (X_i, j) and ran Variant V5 of the algorithm of Figure 5. We halted iteration of the loop in the PPV computation whenever (a) $t \geq 16$ and (b)

the L1-norm of $\mathbf{r}_t$ was less than 10^{-10}. The first condition ensures that every $PPV(v_{i,j}, a_k)$ is non-zero.

With $\gamma = 0.5$, the average relative error (i.e. $|p_\gamma - p|/p$) was 3.7%. The largest relative error[6] among the 100 samples was less than 20%. This is a surprisingly small amount of error. For comparison, we also ran exact inference where each potential $\Phi(X_i = j, X_{i'} = j')$ is perturbed by a randomly-chosen factor a, where a is chosen uniformly at random from $[1 - \epsilon, 1 + \epsilon]$. For $\epsilon = 3 \times 10^{-4}$ the average relative error was 3.4% on the same graph. In many reasonable settings—i.e., if potentials were estimated from data—the error associated with using Variant 5 would not be noticible.

We repeated this experiment with three other randomly-generated graphs of different sizes, and using different values of γ. As expected, errors are smaller with smaller values of γ. The relative error also appears to increase with the size of the graph. These results are summarized in Table 1.

4 Concluding Remarks

Inference in tree-structured MRFs is arguably the most essential and prototypical computation for the subfield of graphical model inference and learning; likewise, personalized PageRank/random walk with restart is an essential and prototypical computation for approaches to data-mining that rely on similarity in structured data. Although widely studied, both practically and theoretically, these two subareas are seldom connected in any concrete way. The primary contribution of this paper is to clarify the connection between MRF inference and similarity measures based on graph walks.

More specifically, in this paper we have established a formal connection between personalized PageRank, a widely-used similarity measure for vertices in a graph, and inference in Markov random fields. We have shown that one can approximate marginal probabilities in a tree-structured pairwise discrete-valued MRF F by performing personalized PageRank computations in the "graph analog" $\hat{G}_u^F$ of the MRF—i.e., an ordinary graph with a similar structure. The "graph analog" used in our construction is quite intuitive: as shown in Figure 2, the graph contains one node v_{ij} for each possible value j that a variable X_i can assume; an edge with weight

[6]Large relative errors always occurred in estimating small probabilities: the largest error was in approximating of the value $Pr(X_i = j) = 3.01 \times 10^{-73}$ with $Pr(X_i = j) \approx 2.52 \times 10^{-73}$.

Graph Size		Relative Error (%)		
$\|V\|$	$\|E\|$	$\gamma = 0.25$	$\gamma = 0.5$	$\gamma = 0.8$
122	480	0.1	0.3	0.9
724	2,888	0.4	1.8	4.9
968	3,864	0.9	3.7	10.4
3326	13,296	1.1	4.6	13.8

Table 1: Performance of Variant 5 on four randomly constructed graphs

$\Phi(X_i = j, X_{i'} = j')$ between nodes for v_{ij} and $v_{i'j'}$, where $\Phi(\cdot, \cdot)$ is the potential function for the MRF; and for every "leaf variable" X_k, a special "anchor node" a_k that is connected to each v_{kj} associated with X_i. Given this construction we show that

- the unnormalized probability or "belief" for a variable's value in F, $Bel_F(X_i = j)$, is approximately proportional to v_{ij}'s similarity to the set of anchor nodes in $\hat{G}_u^F$, where similarity to a set of nodes $\hat{\mathcal{A}}$ is the product of similarity to the individual nodes $a_k \in \hat{\mathcal{A}}$;
- the quality of this approximation is better when γ, the "damping factor" for personalized PageRank, is smaller, becoming a perfect approximation as $\gamma \to 0$;
- experimentally, the approximation is quite good for moderate values of γ (e.g., $\gamma = 0.5$) on certain classes of random MRFs.

Thus the main theorems immediately suggest both an intuitive interpretation of marginal probabilities in an MRF, and an algorithm for MRF inference. Alternatively, one can precompute personal PageRank ranking vectors for each leaf node a_k in an MRF, and then compute $Bel_F(X_i = j)$ as $\prod_{k=1}^{L} APV^{dir}(\hat{G}_u, v_{ij})[a_k]$. This approach is broadly similar to the "message-passing" approach usually used for inference in MRFs, in which "messages" originate in the leaves of the tree and propogate inwards, to be combined at internal nodes.

Our results were developed by analogy to results involving "all-paths similarity" and "directed" versions of all-paths similarity and personalized PageRank—similarity measures that are easier to analyze, but not as commonly used in practice. These intermediate results may be of independent interest, e.g. as pedagogical devices for presenting MRF inference to audiences familiar with graph-walk similarity (or vice versa). Although both MRF inference and graph-walk similarity measures are well-studied formally, the proof of our results are quite accessible, requiring little technical machinery from either subarea.

One reason for the popularity of Bayes networks and other graphical probabilistic models is that they can be implemented with highly parallel "marker passing" schemes [28]—a property that makes the presence of similar inference schemes in the human brain seem somewhat more plausible. Personalized PageRank is also easily parallelized[7] and is in fact quite similar to the class of cognitive models called *spreading activation models* [29]. It is possible that further insight into the neural plausibility of graphical-model inference schemes could be gained by exploiting our reduction of the marker-passing process of belief propogation to the even simpler parallel process of computing personalized PageRank ranking vectors.

One prior work that establishes a connection between graph walks and MRF inference is the "walk-sum" framework of Malioutov *et al* [19]. The walk-sum framework is applied to *Gaussian Markov random fields* (in which each variable X_i takes on a real value $x_i \in \mathcal{R}$) and edge weights indicate covariances, and one of their results is that inference for certain classes of Gaussian graphical models can be implemented by a "walk-sum" process similar to computation of all-paths ranking vectors [19, Appendix A]. Their basic result and construction is broadly similar to the analysis

[7] If each vertex has a processor, one way to compute $PPV(v)$ would be the following. At time 0, "send" activation 1 to processor v. At time $t > 0$, let each processor v' do the following steps: (1) "receive" total activation a from its neighbors; (2) "keep" activation $(1 - \gamma)a$ by adding it to a counter $s[v']$; and (3) "send" activation $\gamma a \mathbf{W}[v', v'']$ to each neighbor v''.

of all-paths similarity in Section 3.1—the main technical difference is that for Gaussian random fields, the construction of "analogous" ordinary graphs is simpler. Malioutov *et al* also define a more general condition of "walk-summability", which holds for certain cyclic Gaussian MRFs, and show that the walk-sum method version of belief propogation will converge for all "walk-summable" networks. However, they do not explore the consequences of using PageRank-style damped walks as we do (i.e., they have no results analogous to our Theorems 2 or 4).

It is to be hoped that many of the well-developed results and techniques from these two subareas can be fruitfully combined to obtain new practical or formal contributions. Another possible point of synergy is between techniques for quickly approximating graph-walk similarities (e.g., [13, 9, 32, 6]) and inference tasks on large MRFs. These are of particular interest because some theoretical approaches to approximate inference in general MRFs (e.g., recent work in inference via self-avoiding walks [34, 15]) reduce the general inference problem in moderate-sized MRFs to inference in larger "computation trees", which are tree-structured MRFs. (One technical obstacle to immediate application of fast approximation techniques for graph-walk similarity is that some approximations ignore very small similarities, which can be important in computing marginal probabilities.) Another obvious area for further study is the performance of the inference method analyzed in Theorem 4 for general MRFs. This method is quite similar to loopy belief propogation [28, 25], but is guaranteed (by Theorem 3) to converge; however, analysis of its limiting accuracy on MRFs that are not trees remains open. One potential avenue of inquiry along these lines is to extend the notion of walk-summability to discrete-valued MRFs.

References

[1] Alekh Agarwal, Soumen Chakrabarti, and Sunny Aggarwal. Learning to rank networked entities. In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 14–23, New York, NY, USA, 2006. ACM Press.

[2] John R. Anderson. *How can the human mind occur in the physical universe?* Oxford University Press, New York, NY, USA, 2007.

[3] Andrey Balmin, Vagelis Hristidis, and Yannis Papakonstantinou. Objectrank: Authority-based keyword search in databases. In Mario A. Nascimento, M. Tamer Özsu, Donald Kossmann, Renée J. Miller, José A. Blakeley, and K. Bernhard Schiefer, editors, *VLDB*, pages 564–575. Morgan Kaufmann, 2004.

[4] G. Bhalotia, A. Hulgeri, C. Nakhe, S. Chakrabarti, and S. Sudarshan. Keyword searching and browsing in databases using BANKS. In *Proceedings of the 18th International Conference on Data Engineering, 2002*, pages 431–440, San Jose, CA, USA, February 2002.

[5] Christopher Bishop, editor. *Pattern Recognition and Machine Learning*. Springer, New York, NY, USA, 2006.

[6] Soumen Chakrabarti. Dynamic personalized PageRank in entity-relation graphs. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 571–580, New York, NY, USA, 2007. ACM Press.

[7] William W. Cohen and Einat Minkov. A graph-search framework for associating gene identifiers with documents. *BMC Bioinformatics*, 2006. To appear. Draft available from http://wcohen.com/postscript/normalize-preprint.pdf.

[8] Robert G. Cowell, Steffen L. Lauritzen, A. Philip David, David J. Spiegelhalter, and David J. Spiegelhater. *Probabilistic Networks and Expert Systems*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1999.

[9] Kroly Csalogny, Dniel Fogaras, Balzs Rcz, and Tams Sarls. Towards scaling fully personalized PageRank: Algorithms, lower bounds, and experiments. *Internet Mathematics*, 2(3):333–358, 2005.

[10] Michelangelo Diligenti, Marco Gori, and Marco Maggini. Learning web page scores by error back-propagation. In *IJCAI*, 2005.

[11] Taher H. Haveliwala. Topic-sensitive pagerank. In *WWW*, pages 517–526, 2002.

[12] Glen Jeh and Jennifer Widom. Simrank: a measure of structural-context similarity. In *KDD*, pages 538–543. ACM, 2002.

[13] Glen Jeh and Jennifer Widom. Scaling personalized web search. In *WWW*, pages 271–279, 2003.

[14] Michael I. Jordan, editor. *Learning in graphical models*. MIT Press, Cambridge, MA, USA, 1999.

[15] K. Jung and D. Shah. Inference in Binary Pair-wise Markov Random Fields through Self-Avoiding Walks. *ArXiv Computer Science e-prints*, October 2006.

[16] R. I. Kondor and J. Lafferty. Diffusion kernels on graphs and other discrete structures. In *Proceedings of the ICML*, 2002.

[17] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th International Conf. on Machine Learning*, pages 282–289. Morgan Kaufmann, San Francisco, CA, 2001.

[18] David Liben-Nowell and Jon M. Kleinberg. The link prediction problem for social networks. In *CIKM*, pages 556–559. ACM, 2003.

[19] Dmitry M. Malioutov, Jason K. Johnson, and Alan S. Willsky. Walk-sums and belief propagation in Gaussian graphical models. *JMLR*, 7:2031–2064, Oct 2006.

[20] Andrew McCallum and Wei Li. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of The Seventh Conference on Natural Language Learning (CoNLL-2003)*, Edmonton, Canada, 2003.

[21] M. Meila and J. Shi. A random walks view of spectral segmentation, 2001.

[22] Sergey Melnik, Hector Garcia-Molina, and Erhard Rahm. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In *ICDE*, pages 117–128. IEEE Computer Society, 2002.

[23] Einat Minkov and William Cohen. Learning to rank typed graph walks: Local and global approaches. In *Proc. of WebKDD-2007*, 2007.

[24] Einat Minkov, William W. Cohen, and Andrew Y. Ng. Contextual search and name disambiguation in email using graphs. In *ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2006.

[25] Kevin Murphy, Yair Weiss, and Michael Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence (UAI-99)*, pages 467–47, San Francisco, CA, 1999. Morgan Kaufmann.

[26] Larry Page, Sergey Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. In *Technical Report, Computer Science department, Stanford University*, 1998.

[27] Christopher R. Palmer and Christos Faloutsos. Electricity based external similarity of categorical attributes. In Kyu-Young Whang, Jongwoo Jeon, Kyuseok Shim, and Jaideep Srivastava, editors, *PAKDD*, volume 2637 of *Lecture Notes in Computer Science*, pages 486–500. Springer, 2003.

[28] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[29] M. Ross Quillian. The teachable language comprehender: a simulation program and theory of language. *Commun. ACM*, 12(8):459–476, 1969.

[30] Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine Learning*, 62(1-2):107–136, 2006.

[31] F. Sha and F. Pereira. Shallow parsing with conditional random fields. In *In Proceedings of HLT-NAACL*, 2003.

[32] Hanghang Tong, Christos Faloutsos, and Jia-Yu Pan. Fast random walk with restart and its applications. In *ICDM*, pages 613–622. IEEE Computer Society, 2006.

[33] Kristina Toutanova, Christopher D. Manning, and Andrew Y. Ng. Learning random walk models for inducing word dependency distributions. In *ICML*, 2004.

[34] Dror Weitz. Counting independent sets up to the tree threshold. In *STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 140–149, New York, NY, USA, 2006. ACM Press.

[35] X. Zhu. Semi-supervised learning with graphs. Technical Report CMU-LTI-05-192, Carnegie Mellon University, 2005.

A Proof of Equation 6

Although substantial technical machinery has been developed in both subareas (Markov random fields and analysis of graph similarities) the proofs of this paper are largely self-contained. The principle exception to this is Equation 6, which claims that

$$Bel_F(X_i = j) = \prod_{b=1}^{B} \sum_{j' \in \mathcal{Y}} Bel_{F_{m_b}}(X_{m_b} = j') \Phi(X_i = j, X_{m_b} = j')$$

This claim can actually be established quite easily, without recourse to graphical model theory. Let us begin by considering a very simple variant of the sort of independencies needed to establish this. Consider sets $\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{R}_1, \mathcal{R}_2, \mathcal{S}$, let $\mathcal{X} \equiv \mathcal{Q}_1 \times \mathcal{R}_1 \times \mathcal{Q}_2 \times \mathcal{R}_2 \times \mathcal{S}$, and let $\mathcal{X}_j \equiv \{\langle q_1, r_1, q_2, r_2 \rangle \in \mathcal{X} : s = j\}$. Define the function $\beta(j)$ as

$$\beta(s) \equiv \sum_{\mathbf{x} \in \mathcal{X}_j} f_1(q_1, r_1) g_1(r_1, s) f_2(q_2, r_2) g_2(r_2, s)$$

where the f_i's and g_i's are arbitrarily functions. Below we will write $\beta(s)$ as $\beta(S = s)$, to help reinforce the similarity of this to computation of "belief" in an MRF. We claim

Proposition 4

$$\begin{aligned} \beta(S = s) &= \sum_{r_1} \sum_{q_1} \sum_{r_2} \sum_{q_2} f_1(q_1, r_1) g_1(r_1, s) f_2(q_2, r_2) g_2(r_2, s) \\ &= \sum_{r_1} \sum_{q_1} f_1(q_1, r_1) g_1(r_1, s) \sum_{r_2} \sum_{q_2} f_2(q_2, r_2) g_2(r_2, s) \end{aligned}$$

The proof of the proposition is immediate, requiring only distributing the common factors across two summations.

From this proposition we can easily generalize to the following:

Lemma 2 *Let B be an integer, let $\mathcal{Q}_1, \ldots, \mathcal{Q}_B$, $\mathcal{R}_1, \ldots, \mathcal{R}_B$ and $\mathcal{S}$ be sets. Define*

$$\mathcal{X}^B = \mathcal{Q}_1 \times \mathcal{R}_1 \times \mathcal{Q}_2 \times \mathcal{R}_2 \times \ldots \mathcal{Q}_B \times \mathcal{R}_B \times \mathcal{S}$$

and let $\mathcal{X}_j^B = \{\langle q_1, r_1, \ldots, q_B, r_B, s \rangle \in \mathcal{X} : s = j\}$. Define

$$\beta_{\mathcal{X}}^B(S = s) \equiv \sum_{\mathbf{x} \in \mathcal{X}_j^B} \prod_{b=1}^{B} f_b(q_b, r_b) g_b(r_b, s) \tag{11}$$

where again the f_b's and g_b's are arbitrarily functions. Then

$$\beta_{\mathcal{X}}^B(S = s) = \prod_{b=1}^{B} \sum_{r_b \in \mathcal{R}_b} \sum_{q_b \in \mathcal{Q}_b} f_b(q_b, r_b) g_b(r_b, s) \tag{12}$$

Proof: If we write Equation 11 as

$$\beta^B_{\mathcal{X}}(S=s) \equiv \sum_{r_1}\sum_{q_1}\cdots\sum_{r_B}\sum_{q_B} f_1(r_1,q_1)g_1(q_1,s)\ldots f_B(r_B,q_B)g_B(q_B,s)$$

we see that we can again distribute the common factors $f_1(r_1,q_1)g_1(q_1,s)$ across most of the sums, yielding

$$\beta^B_{\mathcal{X}}(S=s) \equiv \sum_{r_1}\sum_{q_1} f_1(r_1,q_1)g_1(q_1,s)\sum_{r_2}\sum_{q_2}\cdots\sum_{r_B}\sum_{q_B} f_2(r_2,q_2)g_2(q_2,s)\ldots f_B(r_B,q_B)g_B(q_B,s)$$

We can now distribute out the common factors $f_2(r_2,q_2)g_2(q_2,s)$, and so on: continuing this process repeatedly will yield Equation 12.

Now, consider Equation 6 and define $\mathcal{T}_b$ to be the variables in F_{m_b}, and $\mathcal{R}_b$ to be

$$\mathcal{R}_b \equiv \{\mathbf{r}_b : \ \mathbf{r}_b \text{ is an assignment to the variables in } \mathcal{T}_b - \{X_{m_b}\}\}$$

For $b=1,\ldots,B$, let $\mathcal{Q}_b = \mathcal{Y}$ represent the possible values of X_{m_b}, and let $\mathcal{S}=\mathcal{Y}$ be the possible values of X_i. With a slight abuse of notation we will also let $\mathcal{X} \equiv \langle \mathbf{r_1}, x_{m_1}, \ldots, \mathbf{r_b}, x_{m_b}, x_i\rangle$ and let $\mathcal{X}_j = \{\mathbf{x}\in\mathcal{X} : x_i = j\}$. (Note that $\mathbf{x}\in\mathcal{X}$ is isomorphic to the vectors $\langle x_1,\ldots,x_n\rangle$ that we used in the body of the paper to represent an assignment of values to the variables $X_1,\ldots,X_n$, and $\mathcal{X}_j$ is isomorphic to the set denoted $\mathcal{X}_{ij}$ used in the body of the paper—only the ordering of the variables is changed.) Let E_{Φ_b} be the edges in E_Φ between variables in $\mathcal{T}_b$, and define

$$f_b(\mathbf{r}_b, q_b) \equiv \prod_{(i',i'')\in E_{\Phi_b}} \Phi(X_{i'}=x_{i'}, X_{i''}=x_{i''})$$

where $x_{i'}$ is the value assigned to $X_{i'}$ by $\mathbf{r}$ (if $X_{i'}\in\mathcal{T}_b$) or by q_b (if $X_{i'}=X_{m_b}$). Also define

$$g_b(q_b,s) \equiv \Phi(X_{m_b}=q_b, X_i=s)$$

Substituting these values into Equation 11 gives us

$$\begin{aligned}
Bel_F(X_i=s) &\equiv \sum_{\mathbf{x}\in\mathcal{X}_j}\prod_{b=1}^{B} f_b(q_b,r_b)g_b(r_b,s)\\
&= \prod_{b=1}^{B}\sum_{q_b\in\mathcal{Q}_b}\sum_{\mathbf{r}_b\in\mathcal{R}_b} f_b(q_b,r_b)g_b(r_b,s)\\
&= \prod_{b=1}^{B}\sum_{x_{m_b}\in\mathcal{Y}}\sum_{\mathbf{r}_b\in\mathcal{R}_b}\left(\prod_{(i',i'')\in E_{\Phi_b}}\Phi(X_{i'}=x_{i'},X_{i''}=x_{i''})\right)\Phi(X_{m_b}=x_{m_b},X_i=s)\\
&= \prod_{b=1}^{B}\sum_{x_{m_b}\in\mathcal{Y}}\left(\sum_{\mathbf{r}_b\in\mathcal{R}_b}\prod_{(i',i'')\in E_{\Phi_b}}\Phi(X_{i'}=x_{i'},X_{i''}=x_{i''})\right)\Phi(X_{m_b}=x_{m_b},X_i=s)\\
&= \prod_{b=1}^{B}\sum_{x_{m_b}\in\mathcal{Y}} Bel_{F_{m_b}}(X_{m_b}=x_{m_b})\Phi(X_{m_b}=x_{m_b},X_i=s)
\end{aligned}$$

This concludes the proof of the statement of Equation 6.

Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213

Carnegie Mellon