

Inferring the Maximum Likelihood Hierarchy in Social Networks

Arun S. Maiya

Department of Computer Science
University of Illinois at Chicago
851 S. Morgan, Chicago, IL 60607
Email: amaiya@cs.uic.edu

Tanya Y. Berger-Wolf

Department of Computer Science
University of Illinois at Chicago
851 S. Morgan, Chicago, IL 60607
Email: tanyabw@cs.uic.edu

Abstract—Individuals in social networks are often organized under some hierarchy such as a command structure. In many cases, when this structure is unknown, there is a need to discover hierarchical organization. In this paper, we propose a novel, simple, and flexible method based on maximum likelihood to infer social hierarchy from weighted social networks. We empirically evaluate our method against both simulated and real-world datasets and show that our approach accurately recovers the underlying, latent hierarchy.

I. INTRODUCTION AND MOTIVATION

From troops of baboons to employees in corporations, individuals in a society are often organized by some notion of rank or status. Many animal, bird, and insect species are organized into dominance hierarchies. Similarly, a corporation is structured as an organizational chart with the CEO at the root of the hierarchy. Indeed, recent human imaging studies have revealed that the brain is, in fact, “hard-wired for hierarchy” with specific areas of neural circuitry being associated with social standing and rank [1]. Often times, this underlying social structure is unknown and must be inferred for various applications. For instance, inferences of animal hierarchies help biologists shed light on the species under study, and discovery of the hierarchical organization of terrorist networks is relevant to intelligence analysis and national defense.

Hierarchy may be inferred from interactions among individuals. The hierarchical structure of a population shapes the nature of the social interactions of individuals and, thus, the structure of the underlying social network. With the increased availability of social network data for humans and other social species, it is now possible to understand the connection between the nature of interactions in a network and hierarchy. Surprisingly, there is little work on the automated inference of social hierarchies from networks, and existing approaches often make assumptions about the data that may be unwarranted in some circumstances. Social networks are normally represented by graphs in which the nodes represent individuals and the edges represent associations between the individuals. In this paper, we propose a novel, flexible method to infer social hierarchies in weighted, undirected social networks using maximum likelihood. Specifically, our contributions in this paper are as follows:

- We provide an overview of existing work on the inference

of social hierarchy across a variety of different disciplines from biology to sociology to computer science.

- We propose a formal framework for treating hierarchies as probabilistic, generative models for social networks and show how this framework may be used to measure the likelihood of hierarchical structures given a network.
- We propose a simple optimal algorithm to recover the maximum likelihood hierarchy from a weighted social network.
- We empirically evaluate our approach against both simulated and real-world datasets and show that it accurately recovers the underlying hierarchy.

II. RELATED WORK

Biologists have studied and have attempted to infer social hierarchies since the 1920s when Thorleif Schjelderup-Ebbe discovered that hens were organized by a strict, pecking order [2]. Biologists infer hierarchy by first constructing a dominance encounter matrix showing the number of observed encounters in which individual i dominated individual j . Using this matrix, various approaches, most based on paired comparisons or statistical inference, can be applied to reconstruct the linear dominance hierarchy (e.g. Refs. [3], [4], [5], [6]). However, virtually all of the biological approaches to discovering hierarchy assume the underlying hierarchy is *linear* (i.e. a tree with a branching factor of 1). Furthermore, these approaches assume that the social network under analysis is a directed graph (i.e. the dominance encounter matrix). In many situations, this information is simply not available. Rather, the data available are in the form of an undirected graph representing the social network or a directed graph in which the directed edges may not correspond to dominating interactions (e.g. email networks).

Discovering hierarchical structure in these cases, where the graph is not directed or the hierarchy is not linear, has been a topic of study by sociologists and, more recently, computer scientists and mathematicians. Much of the work in this area employs graph-theoretic centrality measures to reconstruct the hierarchy [7]. Multiple authors such as Memon et al. [8] have proposed using degree and eigenvector centrality to reconstruct the hierarchical structure of terrorist networks under the assumption that higher ranked individuals will have

higher centrality scores. A weighted combination of multiple such graph-theoretic measures was used in an attempt to infer the hierarchical rank of employees in the Enron email network [9]. In [10], a method referred to as canonical analysis of asymmetry was proposed, but, like biological approaches, it is restricted to directed graphs and linear orders. More recently, Kemp et al. [11] have employed Bayesian inference and graph grammars to reconstruct hierarchy among other structures.

Virtually all of the aforementioned works make fixed assumptions regarding the interaction patterns among the individuals in the network with the expectation that these assumptions hold in all cases. For instance, approaches based on graph-theoretic centrality measures assume a direct correspondence between hierarchical position and the centrality measures being considered. We conjecture that the interaction patterns can vary across different types of social networks, and this variability can pose problems for existing approaches. A centrality measure corresponding to hierarchical rank in one network may not necessarily correspond to rank in a different type of network. We propose an approach to the inference of social hierarchy that we show to be more flexible and robust under variable social interaction patterns. We base our approach on a treatment of the underlying hierarchy as a generative model for the network, directly incorporating the notion that an underlying hierarchy, if exists, shapes the structure of the social network.

III. HIERARCHIES AS GENERATIVE MODELS

Consider the CEO of a corporation and his or her position in the organizational chart. Intuitively, the probability of this CEO interacting with immediate subordinates will be higher than the probability of interacting with individuals toward the bottom of the organizational chart, such as the mail room employee. In our work, we make the basic assumption that social interactions among a group of individuals in a hierarchy are influenced by the hierarchical positions of these individuals. As the distance between individuals within a hierarchy grows, we assume the probability of interaction decays. Under this assumption, since the frequency and occurrence of social interactions between two individuals are influenced by their respective positions in the hierarchy, the underlying hierarchy can be considered a *generative model* for the social network. We now provide some formal definitions of social hierarchies, interaction models, and the networks derived from them.

Definition 1 (HIERARCHY): A *hierarchy* is a rooted, directed tree represented by graph, $G_H = (V_H, E_H)$. An edge $(v, w) \in E_H$ denotes that v is dominant over w with v being referred to as the parent and w being referred to as the child. Two nodes with the same parent are referred to as siblings. If a path exists from some node v to some node w , then v is an ancestor of w and w is a descendant of v .

Definition 2 (INTERACTION MODEL): An *interaction model*, M , defines the probability of interaction for each of the $\binom{|V_H|}{2}$ pairs of vertices or individuals in a hierarchy. Let P_B be the *base probability*, which is the highest probability of interaction possible under the model. If a model stipulates

that two nodes either interact rarely or not at all, then the probability of interaction is set to some relatively small, non-zero value to represent noise. Let this value be ϵ .

Definition 3: [PROXIMITY-BASED INTERACTION MODEL] A *proximity-based interaction model* is an interaction model in which the highest rates of interaction are assigned between parents and their children and between siblings. The interaction probabilities for all other pairs of nodes either decay with tree distance or remain constant. Note that this definition does not specify whether it is parents and children or siblings that have the higher interaction probability.

Having defined a *proximity-based interaction model*, we now describe four specific instantiations.

Definition 4 (DIRECT MODEL): In the *Direct Model*, there is a direct correspondence between edges in the hierarchy and edges in the generated network. The probability of interaction between a parent and its children is set to be the base probability, P_B . For the remaining node pairs in the hierarchy, the probability of interaction is set to ϵ .

Definition 5 (DISTANCE MODEL): For pairs of nodes in the hierarchy that have an ancestor-descendant relationship or are siblings, the probability of interaction is proportional to the tree distance (i.e. number of hops) in the hierarchy. All other pairs of nodes are assigned a probability of ϵ .

Definition 6 (MANAGER-DRIVEN MODEL): The *Manager-Driven Model* is identical to the *Distance Model* except the probability of interaction between siblings is changed to be ϵ (i.e. team members interact only through their manager).

Definition 7 (TEAM-DRIVEN MODEL): The *Team-Driven Model* is also similar to the *Distance Model* except the highest rate of interaction, P_B , is assigned to siblings instead of parents-children. For pairs of nodes having an ancestor-descendant relationship, the probability of interaction is proportional to distance.

Several points should be made regarding interaction models. First, although interaction models dictate the probabilities for interactions, they do not specify a fixed definition of an interaction. An *interaction* in an interaction model can be defined as anything from physical proximity recorded through GPS to email exchanges. Second, this list of four interaction models above is not meant to be exhaustive. These are simply the models defined, chosen, and evaluated in the present work. Finally, it should also be noted that, in three of the four models described (all, but the *Direct Model*), interaction probabilities for ancestor, descendant, and/or sibling relationships are a function of tree distance with probabilities decaying with distance. There are many different functions that may be chosen for the decay rate in these models. For this paper, we evaluate two rates of decay: exponential and inverse linear. The exponential decay of interaction probability is defined as $Pr[(v, w)] = (P_B)^{dist(v, w)}$ where $Pr[(v, w)]$ is the interaction probability between nodes v and w

and $dist(v, w)^1$ is tree distance. The inverse linear decay of interaction probability is defined as $Pr[(v, w)] = \frac{P_B}{dist(v, w)}$.

Definition 8 (HIERARCHY-DERIVED NETWORK): A *hierarchy-derived, social network* is a weighted graph $G_N = (V_N, E_N)$ where $V_N = V_H$ and every edge $(v, w) \in E_N$ has a weight proportional to the probability dictated by some interaction model, M , associated with the hierarchy, G_H . This weight, for example, might be the frequency of social interactions between individuals over some time period.

We conclude this section with a formal definition of the problem of inferring hierarchy from a social network..

Definition 9: Given G_N , infer both G_H and M that generated G_N .

This is a general definition for the problem of inferring the *true* underlying hierarchy. In the absence of ground truth (which necessitates the problem of inference), we look for the most *likely* hierarchy.

IV. METHOD

A. Overview

At its core, our approach to the problem employs the use of model selection with maximum likelihood estimation. We treat each of the four models above as *candidate models*. Given the data G_N , under each candidate model, we find the hierarchy with the maximum log-likelihood using a greedy, iterative algorithm called *Hi-GreeMax* (see Algorithm 1) and record its likelihood score. We, then, rank each candidate model by the log-likelihood score of the maximum likelihood hierarchy it produces. Finally, we take the highest ranked candidate model as the inferred model M and the hierarchy with which it is associated as our inferred hierarchy G_H .

B. Measuring the Likelihood of a Hierarchy

Given a candidate model M , for any two individuals v and w placed in positions in a hierarchy, the log-likelihood (LL) for the weight between the pair can be computed as follows:

$$LL(v, w) = \alpha \log(Pr[(v, w)]) + \beta \log(1 - (Pr[(v, w)])) \quad (1)$$

where

$$Pr[(v, w)] = \text{probability of interaction} \quad (2)$$

under given candidate model

$$\alpha = \text{weight}(v, w) \quad (3)$$

$$\beta = \frac{1}{P_B} \max_{(v_1, v_2) \in E_N} \{ \text{weight}(v_1, v_2) \} - \text{weight}(v, w) \quad (4)$$

¹Nodes v and w here are assumed to have either an ancestor-descendant relationship or a sibling relationship.

The log-likelihood for an entire hierarchy, then, is:

$$\ell(G_H | G_N, M) = \sum_{v, w \in V_N, v \neq w} LL(v, w) \quad (5)$$

This likelihood computation can be viewed in terms of a Bernoulli process or coin-flipping experiment. In our model, we consider interactions as occurring over a number of trials with each interaction occurring with some probability. This is not to say that the actual interactions or associations proceed in this fashion. This is merely how the data are modeled in our approach. During each trial, between any two nodes, an interaction can either occur or not occur (analogous to heads vs. tails). The edge weights, then, are treated as the number of successes (e.g. number of heads) shown as α in the equations above. In order to measure the likelihood, we require the number of trials in which *no* interaction occurred (e.g. number of tails), which is the total number of trials *minus* the number of trials interactions occurred. The total number of trials, however, including the number of trials in which no interaction took place, may be unknown. Under our model, the total number of trials can be inferred by dividing the highest edge weight by the base probability P_B . The number of trials where no interaction took place, shown as β in the equations, is taken to be the total number of trials minus the edge weight (i.e. the number of trials interactions occurred). One might also view our generative model for networks as a variation of the Erdos-Renyi random graph model in which there is a non-uniform probability assigned to edge creation [12].

C. Finding the Maximum Likelihood Hierarchy

For a network of n nodes, there are n^{n-1} possible rooted, labeled trees (see Chapter 9 in [13]). A simple, brute-force search for the maximum likelihood hierarchy over all trees of n nodes, therefore, is infeasible. We propose a simple greedy iterative algorithm *Hi-GreeMax* and show that it finds the maximum likelihood hierarchy. The *Hi-GreeMax* algorithm builds the hierarchy by iteratively placing nodes in positions that maximize the likelihood (see Algorithm 1). The algorithm accepts as input the network, G_N , and a candidate interaction model, M_C . It begins by selecting a single node, *seed*, around which the entire hierarchy is constructed. In Algorithm 1, the node *seed* is chosen as the node with highest weighted degree centrality, but this is not a requirement. It should be noted that *seed* does not necessarily have to be (and in many cases will not be) the root of the hierarchy. All the remaining nodes are chosen such that they maximize the total sum of interactions with the nodes already chosen.

Once the first three nodes are selected, the *FindMaxTriad* procedure performs a quick, exhaustive search over the ten possible 3-node hierarchy configurations (referred to as triads) to find the configuration which maximizes the likelihood. These triads, shown in Figure 1, include the nine, rooted labeled trees and a tenth configuration where the three nodes are assumed to be siblings of one another. The remaining nodes are positioned in the hierarchy by the *PlaceInMaxLikelihoodPosition* procedure. *PlaceInMaxLikelihoodPosition* tries

Algorithm 1 Hi-GreeMax

1: **Input:**
 Graph $G_N = (V_N, E_N)$
 Candidate Model M_c
2: $seed \leftarrow \arg \max_v \sum_{w \in V_N} weight(v, w)$
3: $placed \leftarrow \{seed\}$
4: $v_2 \leftarrow \arg \max_v [\sum_{w \in placed} weight(v, w)]$
5: $placed \leftarrow placed \cup \{v_2\}$
6: $v_3 \leftarrow \arg \max_v [\sum_{w \in placed} weight(v, w)]$
7: $placed \leftarrow placed \cup \{v_3\}$
8: $FindMaxTriad(seed, v_2, v_3)$
9: **while** $|placed| \leq |V_N|$ **do**
10: $v_n \leftarrow \arg \max_v [\sum_{w \in placed} weight(v, w)]$
11: $PlaceInMaxLikelihoodPosition(v_n)$
12: $placed \leftarrow placed \cup \{v_n\}$
13: **end while**

positioning the new node as either the new root of the partially constructed hierarchy or a child of one of the existing nodes². The position maximizing the likelihood of the hierarchy is always chosen. This process continues until all nodes are positioned and the maximum likelihood hierarchy is obtained.

As mentioned previously in Section III, a given candidate interaction model has additional parameters P_B and ε (the base probability and noise probability, respectively). The parameter spaces for both P_B and ε are appropriately discretized and quickly searched. The hierarchy returned by *Hi-GreeMax*, then, is the maximum likelihood hierarchy produced for a given candidate model across the discretized, parameter space of both P_B and ε . We now state Theorem 1, the proof for which shows *Hi-GreeMax* is optimal.

Theorem 1: Given a proximity-based interaction model as defined in Definition 3 and a network G_N of n vertices, *Hi-GreeMax* finds the expected maximum likelihood hierarchy. Theorem 1 can be shown through a proof by induction. We omit the proof due to space constraints.

V. EXPERIMENTAL EVALUATION

We validated our approach on both simulated and real-world datasets. For real-world datasets, we infer the hierarchies of the George W. Bush administration (2001-2005) and the Barack H. Obama administration (2009-2013). In all cases, our method recovered the true, underlying hierarchy.

A. Simulated Datasets

We first randomly generated a hierarchy, G_H , of n nodes. This was done by specifying the maximum branching factor and the maximum number of nodes for the tree. Then, beginning from the root node, we probabilistically grew the tree in an iterative fashion. While growing the tree, we selected the number of children for each node uniformly at random such that the number of children did not exceed the maximum

²If the triad found by *FindMaxTriad* is the tenth configuration, *PlaceInMaxLikelihoodPosition* also tries positioning the new node as an additional sibling in the partially constructed hierarchy.

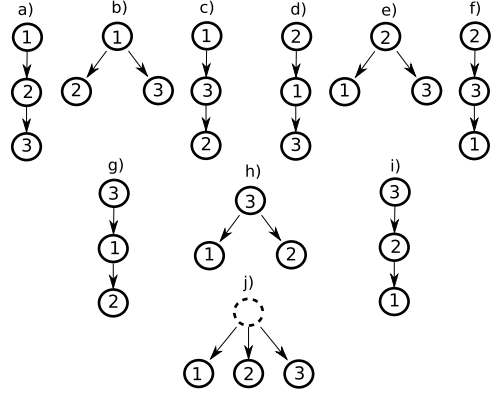


Fig. 1. The ten possible triad configurations.

branching factor specified. This process generated mostly non-complete trees. We also generated complete trees by ensuring each non-leaf node was assigned a number of children equal to the specified branching factor. For our experiments, we generated hierarchies of up to approximately $n = 250$ nodes using branching factors of up to 8.

Second, we selected some interaction model M (of those described in Section III) and randomly generated a network, G_N , based on that model. This was accomplished by first specifying a number of timesteps, T . Then, at each timestep, for each of the $\binom{V_H}{2}$ pairs of nodes, an interaction was created with probability specified by the interaction model, M . The final edge weight between each pair of nodes, then, was set to be the total number of interactions created over the T timesteps. For our experiments, we set $T = 500$.

Upon generating the network G_N , we executed our algorithm to find the maximum likelihood hierarchy under each candidate interaction model. Under our paradigm, the candidate model producing the hierarchy with the optimal likelihood should be the model which generated the network in the first place (i.e. M), thereby, providing validation of our approach. We performed the aforementioned steps for all interaction models, for various hierarchical configurations and for various values of n of up to 250.

In all cases, we were able to reconstruct the original hierarchy perfectly and also infer the generative model, M . Results for a 63 node hierarchy are shown in Figure 2. Each group of bars represents results against a network produced by a particular generative model. Each individual bar shows the negative log-likelihood of the hierarchy produced by each candidate model. For ease of illustration, we graph the *negative* log-likelihood (lower scores are better) instead of the log-likelihood. For instance, the first bar in the first group of bars shows the negative log-likelihood of the hierarchy produced by the *Direct Candidate Model* against a network produced by the *Direct Generative Model*. As expected, the candidate model represented by the best (lowest) bar always matches the generative model. This trend holds for each group in the figure. Due to space constraints, we only include the results

for the 63 node experiment, but results are similar for larger and smaller networks under both exponential and inverse linear rates of decay. This consistency across experiments provides validation for our approach.

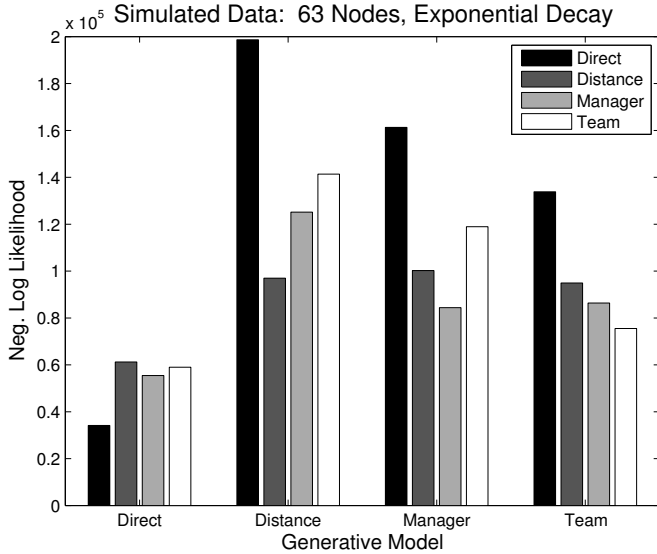


Fig. 2. Negative Log-Likelihood scores for each candidate model against each generative model. The candidate model with the best (i.e. smallest) score in each group is always the same as the generative model and also reconstructs the original hierarchy perfectly.

B. The George W. Bush Hierarchy

A network of associations among 13 members of the George W. Bush administration (2001-2005) was obtained from Ref. [11]. In this network, Google searches were performed on the string “x told y”, where x and y are names of individuals in the administration. The weight of an edge between x and y is, then, the search result count. We converted this directed network into an undirected network by setting the weight of each edge (x, y) to be the number of page counts for “x told y” plus the number of page counts for “y told x”. We are in no way claiming that proximity of individuals’ names on web pages are representative of real social interactions between the individuals. It is our contention, however, that, given that the occurrence and frequency of an association are influenced by the underlying hierarchy, this latent hierarchy can be inferred.

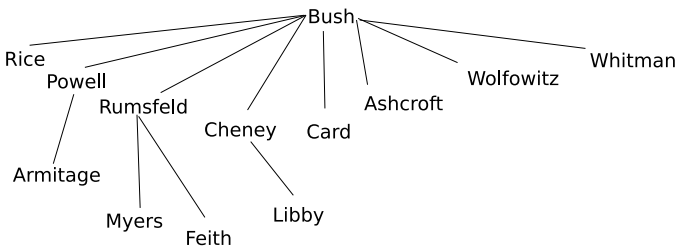


Fig. 3. The inferred hierarchy for the Bush administration dataset produced by the *Direct Model*.

Figure 3 shows the inferred Bush hierarchy produced by the *Direct Model*. As can be seen, this inferred hierarchy closely matches the actual reporting relationships of the Bush administration. The most salient misplaced link is between Bush and Wolfowitz. Wolfowitz, whose position at the time was Deputy Secretary of Defense, should be shown as reporting to Defense Secretary Rumsfeld instead of Bush. In the underlying weighted network, however, the weight between Wolfowitz and Rumsfeld is only 2 whereas the weight between Wolfowitz and Bush is 132. Given the data, it would be impossible for any algorithm to deduce the correct link in this case.

Interestingly, the interaction model producing the best likelihood, as shown in Table I, is the *Team Model* with exponential decay. The hierarchy produced by the *Team Model*, however, does not closely match the real reporting relationships as the *Direct Model* does. Why, then, should the *Team Model* exhibit a better likelihood? We have identified a number of reasons for this model exhibiting the best likelihood some of which have to do with problems relating to the collection process of the dataset itself (e.g. name ambiguity and pre-processing issues). For instance, we surmise that the association frequency between Bush and Powell is inflated due to name ambiguity. In addition to serving as Secretary of State under George W. Bush, Powell also served as Chairman of the Joint Chiefs of Staff under the first President Bush (George H. W. Bush, U.S. President from 1989-1993). We address these and other issues in the next section.

C. The Barack H. Obama Hierarchy

We collected a network of associations among 14 members of the Obama administration³. To address the aforementioned data collection issues present in the Bush dataset, we collected the data following a different process. First, instead of using search strings of the form “x told y”, we used the Google word proximity operator to find the number of pages where the names were within two words of each other. Second, to avoid issues related to name ambiguity and the inclusion of older pages not directly relevant to the administration (e.g. documentation of the Obama-Clinton rivalry during the 2008 Democratic primary), we included current job titles in the search. Finally, we normalized search counts by dividing each page count by $\min(count(x), count(y))$ where $count(\cdot)$ is the page count for a single name.

Table II shows the results. The candidate model producing the best log-likelihood of -2032 is the *Manager Model* with inverse linear decay. The hierarchy produced by this model, which again closely matches the actual reporting relationships, is shown in Figure 4. As with Wolfowitz in the Bush dataset, Flournoy, Under Secretary for Defense Policy, is incorrectly linked to Obama instead of Defense Secretary Gates. The remaining reporting relationships, however, are accurate and closely match the actual Obama administration, as shown in Figure 4. Finally, the candidate model exhibiting the worst

³Collected on April 28, 2009. Note that the revised collection process for this dataset mitigated issues such as name ambiguity, but did not completely eliminate them.

	Direct	Distance		Manager		Team	
		Exponential	Inv. Linear	Exponential	Inv. Linear	Exponential	Inv. Linear
Log-Likelihood	-64179	-75405	-91251	-69528	-66559	-48736	-62832
Inferred P_B	.45	.2	.4	.5	.45	.3	.95
Inferred ε	.001	.001	.001	.001	.001	.001	.001

TABLE I
PERFORMANCE OF EACH INTERACTION MODEL AGAINST THE BUSH HIERARCHY

	Direct	Distance		Manager		Team	
		Exponential	Inv. Linear	Exponential	Inv. Linear	Exponential	Inv. Linear
Log-Likelihood	-2145	-2200	-2307	-2384	-2032	-2441	-2443
Inferred P_B	.35	.25	.25	.35	.35	.45	.25
Inferred ε	.01	.01	.01	.01	.01	.01	.01

TABLE II
PERFORMANCE OF EACH INTERACTION MODEL AGAINST THE OBAMA HIERARCHY

likelihood was the *Team Model*. The hierarchy produced by the *Team Model* (with both inverse linear and exponential decay) was *not* consistent with the actual Obama hierarchy and did not account for the data well at all.

By mitigating issues such as name ambiguity, the previously mentioned discrepancies resulting from the Bush dataset are eliminated, and the candidate model producing the best likelihood (i.e. *Manager Model*) is correctly associated with the hierarchy most closely matching the actual reporting relationships in the Obama administration. Both the model M and the hierarchy G_H are inferred with accuracy, which is consistent with our results for the simulated datasets.

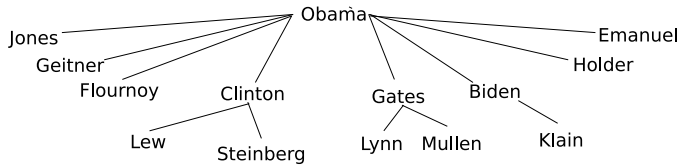


Fig. 4. The inferred hierarchy for the Obama administration dataset produced by the *Manager Model with Inverse Linear Decay*.

VI. CONCLUSION

We have proposed a novel method for inferring both social hierarchies from networks and the generative interaction models that may give rise to the networks. We proposed four models of generating a social network from hierarchical structure. Our method optimally infers the maximum likelihood hierarchy in a given network using a simple greedy algorithm. We have validated our approach on both synthetic and real-world datasets and have shown that our method applies in a more general context than previously proposed techniques. It is the only approach that can infer hierarchy from any undirected weighted network. Moreover, it is the only approach that identifies the most likely style of interactions among individuals in the hierarchy. Our results indicate that hierarchies can be inferred from associations among entities

in a network, given that the occurrence and frequency of these associations are influenced by the underlying hierarchy. In the future, we plan to extend our generative interaction models to include variable probabilities of interactions that depend on the levels of the individuals in the hierarchy.

REFERENCES

- [1] C. F. Zink, Y. Tong, Q. Chen, D. S. Bassett, J. L. Stein, and A. Meyer-Lindenberg, "Know your place: Neural processing of social hierarchy in humans," *Neuron*, vol. 58, no. 2, pp. 273–283, April 2008.
- [2] P. G. Perrin, "Pecking order 1927-54," *American Speech*, vol. 30, no. 4, pp. 265–268, 1955.
- [3] H. Devries, "Finding a dominance order most consistent with a linear hierarchy: A new procedure and review," *Animal Behaviour*, vol. 55, no. 4, pp. 827–843, April 1998.
- [4] H. De Vries, "Finding an appropriate order for a hierarchy: A comparison of the I&SI and the BBS methods," *Animal Behaviour*, vol. 59, no. 1, pp. 239–245, January 2000.
- [5] K. A. Jameson, M. C. Appleby, and L. C. Freeman, "Finding an appropriate order for a hierarchy based on probabilistic dominance," *Animal Behaviour*, vol. 57, no. 5, pp. 991–998, May 1999.
- [6] E. S. Adams, "Bayesian analysis of linear dominance hierarchies," *Animal Behaviour*, vol. 69, no. 5, pp. 1191–1201, May 2005.
- [7] S. Wasserman, K. Faust, and D. Iacobucci, *Social Network Analysis: Methods and Applications (Structural Analysis in the Social Sciences)*. Cambridge University Press, November 1994.
- [8] N. Memon, H. L. Larsen, D. L. Hicks, and N. Harkiolakis, "Detecting hidden hierarchy in terrorist networks: Some case studies," in *PAISI, PACCF and SOCO '08: Proceedings of the IEEE ISI 2008 PAISI, PACCF, and SOCO international workshops on Intelligence and Security Informatics*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 477–489.
- [9] R. Rowe, G. Creamer, S. Hershkop, and S. J. Stolfo, "Automated social hierarchy detection through email network analysis," in *WebKDD/SNA-KDD '07: Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*. New York, NY, USA: ACM, 2007, pp. 109–117.
- [10] L. C. Freeman, "Uncovering organizational hierarchies," *Comput. Math. Organ. Theory*, vol. 3, no. 1, pp. 5–18, 1997.
- [11] C. Kemp and J. B. B. Tenenbaum, "The discovery of structural form," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, no. 31, pp. 10687–10692, July 2008.
- [12] P. Erdos and A. Renyi, "On the Evolution of Random Graphs," *Publ. Math. Inst. Hung. Acad. Sci.*, vol. 5, pp. 17–61, 1960.
- [13] K. H. Rosen and J. G. Michaels, *Handbook of discrete and combinatorial mathematics*. CRC Press, 1999.