Multi-layer graph analytics for social networks

Brandon Oselio, *Student Member, IEEE,* Alex Kulesza, *Member, IEEE,* Alfred O. Hero, III, *Fellow, IEEE*

***Abstract*—Modern social networks frequently encompass multiple distinct types of connectivity information; for instance, explicitly acknowledged friend relationships might complement behavioral measures that link users according to their actions or interests. One way to represent these networks is as multi-layer graphs, where each layer contains a unique set of edges over the same underlying vertices (users). Edges in different layers typically have related but distinct semantics; depending on the application, multiple layers might be used to reduce noise through averaging, perform multifaceted analyses, or a combination of the two. However, it is not obvious how to extend standard graph analysis techniques to the multi-layer setting in a flexible way. In this paper we develop latent variable models and methods for mining multi-layer networks for connectivity patterns based on noisy data.**

***Index Terms*—Hypergraphs, multigraphs, mixture graphical models, Pareto optimality**

Multi-layer networks arise naturally when we have more than one source of connectivity information for a group of users. In a social networking context, we often have knowledge of direct communication links, i.e., *relational* information. However, we might also derive *behavioral* relationships based on user actions or interests. The question that this paper attempts to address is how to deal with these multiple layers of a social network when attempting to perform tasks like inference, clustering, and anomaly detection.

We propose a generative hierarchical latent-variable model for multi-layer networks, and show how to perform inference on its parameters. Using techniques from Bayesian Model Averaging [1], we conditionally decouple the layers of the network using a latent selection variable; this makes it possible to write the posterior probability of the latent variables given the multi-layer network. The resulting mixture can be viewed as a scalarization of a multi-objective optimization problem [2], [3], [4]. When the posterior probability functions are convex, the scalarization of the multiobjective problem is both optimal and consistent with the Bayesian context [2], [5].

We then step back from the Bayesian setting and discuss how multi-objective optimization can be used to perform MAP estimation of the desired latent variables. Using the concept of Pareto optimality [4], we can define an entire front of solutions; this allows a user to define a preference over optimization functions and tune the algorithm accordingly. The result is a level of supervised optimization and inference that still utilizes the structure of multi-layer networks.

We perform experiments on a simulated example, showing that our method yields improved clustering performance in noisy conditions. We discuss how our framework can be combined with existing models, and describe the details of this process for the dynamic stochastic block model (DSBM) [6], which captures a variety of complex temporal network phenomena. Finally, we apply the multi-layer DSBM to a real-world data set drawn from the ENRON email corpus.

The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109, USA. Tel: 1-734-763-0564. Fax: 1-734-763-8041. Emails: {boselio, kulesza, hero}@umich.edu.

This work was partially supported by ARO grant #W911NF-12-1-0443.

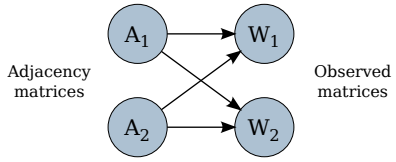

Fig. 1. Simple graphical model ($L = 2$). In general, each observation matrix may be influenced by multiple adjacency matrices.

I. Multi-layer networks

A multi-layer graph $G = (\mathcal{V}, \mathcal{E})$ comprises vertices $\mathcal{V} = \{v_1, \ldots, v_p\}$, common to all layers, and edges $\mathcal{E} = (\mathcal{E}_1, \ldots, \mathcal{E}_L)$ in L layers, where $\mathcal{E}_i$ is the edge set for layer i. We write $A_i \in \{0,1\}^{p \times p}$ to denote the adjacency matrix of layer i: $[A_i]_{uv} = \mathbb{I}[(u,v) \in \mathcal{E}_i]$.

We will assume that the data observed in practice are noisy reflections of this true underlying multi-layer graph, and we denote by $W_i \in \mathbb{R}^{p \times p}$ the observed adjacency weight matrix. In some cases W_i might be binary, reflecting merely the presence or absence of an observed connection—for instance, whether two users were seen to communicate. In other settings, such as measuring temporal or content correlation scores between users, the entries of W_i could be real-valued. Note that the observed matrix W_i may depend on A_j for $i \neq j$; see Figure 1.

II. Hierarchical model description

We wish to estimate $A_1, \ldots, A_L$ given the observations $W_1, \ldots, W_L$. Using standard parametric methods this will require computing posterior distributions of $A_1, \ldots, A_L$, which may be quite complex since the layers are coupled.

Instead, we propose a modified hierarchical model that simplifies the inference procedure. For simplicity, let us specialize to the case where $L = 2$. (For instance, imagine the setting described in the introduction: one layer of the network represents the observed extrinsic relationships between users, and the other their correlated intrinsic behaviors.)

We first introduce a latent variable denoted Y (see Figure 2) that allows the model to continue to express coupling between layers while conditionally *de*coupling their posterior distributions:

$$P(W_1, W_2 | A_1, A_2, Y) = P(W_1 | A_1, Y) P(W_2 | A_2, Y) \ . \quad (1)$$

Since the variables $A_1, \ldots, A_L$ are now intermediaries between Y and the observed weight matrices, we will simplify by collapsing them into $W_1, \ldots, W_L$ and simply inferring Y

978-1-4673-3146-3/13/$31.00 ©2013IEEE

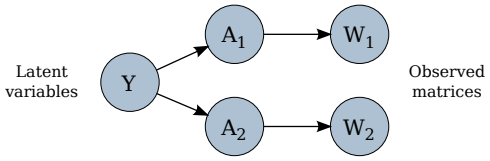

Fig. 2. Latent variable model. The latent variable Y determines the distributions of the adjacency matrices and, through them, the observation matrices.

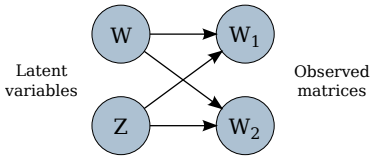

Fig. 3. Model with similarity matrix and selection variable. W and Z take the place of Y, and the adjacency matrices have been collapsed.

itself. (If desired, we can reconstruct $A_1, \ldots, A_L$ later once we know the distribution of Y.) Decomposing $Y = (W, Z)$, we end up with the graphical model in Figure 3, where $W \in \mathbb{R}^{p \times p}$ is a latent adjacency (or similarity) matrix describing the underlying connections between vertices, and $Z \in \{1, 2\}$ is a model selection variable with $P(Z = 1) = \alpha$ and $P(Z = 2) = 1 - \alpha$.

Here we are making the implicit assumption that there is a common connectivity structure W that informs all layers of the network; due to the different attributes of each layer, they may reveal this underlying structure in different ways, or obfuscate it altogether. In a sense the model produces observed matrices that correspond to multiple views of the latent variable W. The model selection variable Z will decouple the posterior distribution of W given both layers into a weighted sum of marginalized posteriors given each individual layer.

The distributions $P(W_1|W, Z)$ and $P(W_2|W, Z)$ are in general task-dependent (e.g., they could be Gaussian, Wishart, Bernoulli, etc.), but we will make the simplifying assumption that Z acts as a selector variable, so that W and W_1 are conditionally independent given $Z = 2$, and likewise W and W_2 are conditionally independent when $Z = 1$. Formally, using the notation P_z to denote conditioning on $Z = z$, we have

$$P_2(W_1|W) = P_2(W_1) \tag{2}$$
$$P_1(W_2|W) = P_1(W_2) \ . \tag{3}$$

We are interested in the posterior distribution of the latent variable W given the observed variables W_1, W_2:

$$\begin{aligned} P(W|W_1, W_2) = \xi P(W|W_1, W_2, Z = 1) \\ + (1 - \xi) P(W|W_1, W_2, Z = 2) \ , \end{aligned} \tag{4}$$

where $\xi = P(Z = 1|W_1, W_2)$. Let's consider the first term. We have

$$P(W|W_1, W_2, Z = 1) = \frac{P(W) P_1(W_1|W) P_1(W_2)}{\sum_{\hat{W}} P(\hat{W}) P_1(W_1|\hat{W}) P_1(W_2)} \ . \tag{5}$$

Since $P_1(W_2)$ does not depend on W, it factors out of the sum in the denominator and cancels; thus we have

$$P(W|W_1, W_2) \tag{6}$$
$$= \xi \frac{P(W) P_1(W_1|W)}{P_1(W_1)} + (1 - \xi) \frac{P(W) P_2(W_2|W)}{P_2(W_2)} \tag{7}$$
$$= P(W) \left[\gamma_1 P_1(W_1|W) + \gamma_2 P_2(W_2|W) \right] \ , \tag{8}$$

where $\gamma_1 = \xi / P_1(W_1)$ and $\gamma_2 = (1 - \xi) / P_2(W_2)$ are constants with respect to W. If we assume the prior on W is uniform, then the MAP value of W is also the maximum likelihood estimate, and can be written as

$$\hat{W} = \text{argmax}_W \left[\gamma_1 P_1(W_1|W) + \gamma_2 P_2(W_2|W) \right] \ . \tag{9}$$

For example, assume that both $P(W_1|W)$ and $P(W_2|W)$ are distributed as isometric Gaussians, i.e.,

$$P(W_1|W) = \mathcal{N}(W, \sigma_1^2 I_p) \tag{10}$$
$$P(W_2|W) = \mathcal{N}(W, \sigma_2^2 I_p) \ . \tag{11}$$

Then the solution to Equation 9 has the form

$$\hat{W} = \beta W_1 + (1 - \beta) W_2 \tag{12}$$

for some $0 \leq \beta \leq 1$.

The above describes not only one MAP estimate of W, but rather a family of MAP estimates based on the priors assigned to each model by α (which affects ξ and γ in turn). Qualitatively, this can be viewed as a relative confidence measure on the layers; if W_1 is more trustworthy than W_2, then the best choice of α would be greater than 0.5.

III. Pareto summarizations

Of course, in practice it may be difficult to effectively set the prior α directly; as a result it might be more useful to generate the entire family of possible solutions and then choose among them afterward. We can view this procedure as a special case of a general framework that offers more flexibility in the inference and estimation procedure.

Let us consider a straightforward multi-objective optimization problem

$$\hat{W} = \text{argmin}_W [f_1(W), f_2(W)] \ . \tag{13}$$

For the model derived in the previous section, we have $f_1(W) = -P_1(W_1|W)$ and $f_2(W) = -P_2(W_2|W)$. One potential approach to the multi-objective optimization problem above is *scalarization* of the two objective functions, so that the new problem to be solved is

$$\hat{W} = \text{argmin}_W \gamma f_1(W) + (1 - \gamma) f_2(W) \ . \tag{14}$$

This view leads to the objective in Equation 9.

However, this can be a somewhat naïve approach to this optimization problem, as potentially valuable solutions may be discarded. A more general notion is Pareto optimality. A solution to a multi-objective optimization problem is said to be weakly Pareto optimal (or weakly non-dominated) if it is not possible to improve any objective function without worsening some other objective function [2], [3]. More formally, we say that a solution W dominates a solution W' if $f_i(W) \leq f_i(W')$ for every objective function f_i and there exists some j such that

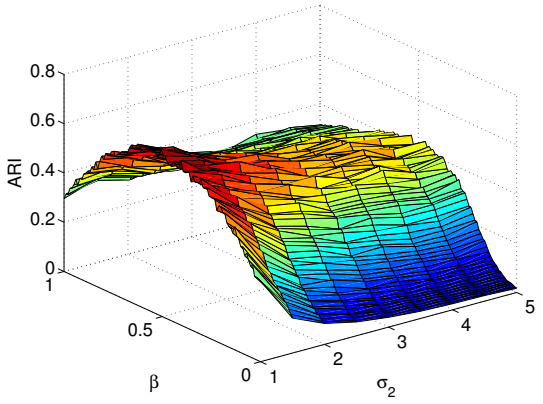

Fig. 4. Clustering simulation. This surface plot shows the ARI for different values of σ_2 and β. Note that in all cases, β that is around 0.5 tends to produce the best clustering.

$f_j(W) < f_j(W')$. The first Pareto front is the set of weakly non-dominated points.

In general, the scalarization technique described above identifies a subset of Pareto optimal points. This subset is complete in some cases; for instance, if the solution space is a convex set and the individual objective functions are convex functions, scalarization gives the full Pareto front [5]. However, when such convexity conditions are not met, the scalarization technique yields an incomplete family of solutions. In our setting, the posterior distributions in Equation 13 are frequently non-convex. Thus, by employing the concept of Pareto optimality, we are extending our list of possible optimal solutions, and generalizing the MAP estimate of Equation 9.

IV. Simulation Example

We use simulations to show that clustering of nodes in a weighted graph can be improved using the MAP estimate of W. Two random graphs with 500 nodes are constructed with 10 known clusters. The weights between nodes in the same cluster are normally distributed as $\mathcal{N}(5, 0.5)$, and weights between nodes that are not in the same cluster are normally distributed as $\mathcal{N}(4.7, 0.5)$. Both layers come from this underlying similarity structure, but are corrupted with i.i.d. Gaussian noise with zero mean and different variances σ_1 and σ_2. For different choices of β, the networks are clustered using a normalized-cut spectral clustering algorithm, and the Adjusted Rand Indices (ARI) [7] are computed. For each of several different levels of variance, this experiment is run 50 times, and the results are averaged. Figure 4 shows a plot of the results, and Table I reports optimal values of β. These results show that using Equation 9 to estimate the mixture of networks improves the clustering. Note that even with unequal variance, optimal β is consistently near 0.5.

TABLE I
Variances and ARI Scores

σ_1	σ_2	Max ARI	β
1	1	0.6782	0.5051
1	1.5	0.6199	0.5253
1	2	0.5828	0.4343
1	2.5	0.5514	0.5051
1	3	0.5073	0.4545
1	3.5	0.4878	0.4848
1	4	0.4876	0.5253
1	4.5	0.4635	0.5354
1	5	0.4429	0.4646

V. ENRON Example

We next look at the ENRON email data set[1]. This data set consists of approximately half a million email messages sent or received by 150 senior employees of the ENRON Corporation. These emails were made publicly available as a result of the SEC investigation of the company in 2002, and constitute one of the largest publicly available email repositories.

To explore *dynamic* multi-level structure, we create two layers from the ENRON dataset over a series of time periods. The information that builds the layers is chosen so that one layer represents the extrinsic, "relational" information between users, and the other represents intrinsic, "behavioral" information between users.

First, a *relational* network is recovered from the headers of emails by identifying the sender and receiver(s) of each message, including Cc and Bcc recipients. For each week in the dataset, a separate network of employees is constructed from the emails sent during that week. A second set of *behavioral* networks are recovered using the contents of email messages. On the same weekly basis the contents of all emails originating from each user are combined to form long "documents", for which term frequency-inverse document frequency (TF-IDF) scores are calculated [8]. Using the vector of TF-IDF scores for each user, we then apply the standard metric of cosine similarity and obtain a symmetric matrix that forms the second observed layer for a given week.

In order to perform inference on this node set, we employ the dynamic stochastic block model (DSBM) [6]. This method infers the probabilities of connection inside and outside of communities, and treats members of the same community as statistically identical. It then propagates this model through time using an extended Kalman filter structure. Since we wish to use this framework, it is necessary to transform the weighted edge network into a binary network. To do this, the similarity scores are thresholded. To be roughly consistent with the density of the relational network, we keep the top 15% greatest correlations between users at each time step, setting all other connections to 0. This allows us to create networks of similar sparsity level. The above procedure yields a two-layer binary dynamic network that we can use to obtain insight into the structural dynamics of the ENRON data. For the DSBM structure, we group employees by their role in the company (CEO, President, Director, etc.).

Combining the two networks as in Section II, we run the DSBM for different levels of the mixing parameter α. Because of the use of binary networks in this example, the α parameter is used as the probability that the combined data will choose to use the relational network when the two layers disagree with

[1] http://www.cs.cmu.edu/ enron

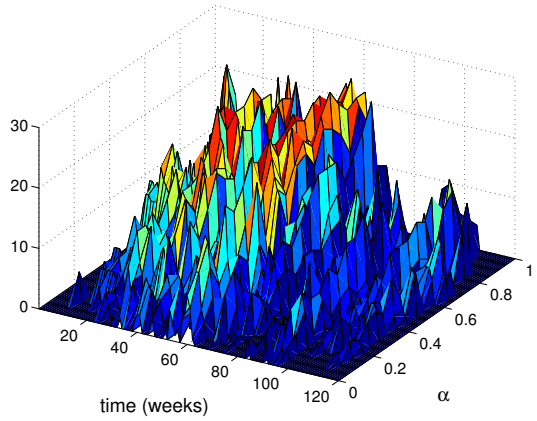

Fig. 5. Betweenness centrality for directors. This centrality is a measure of how connected a node is to the rest of the network. Larger centrality scores often occur for intermediate values of α, particularly between time 95 and 115.

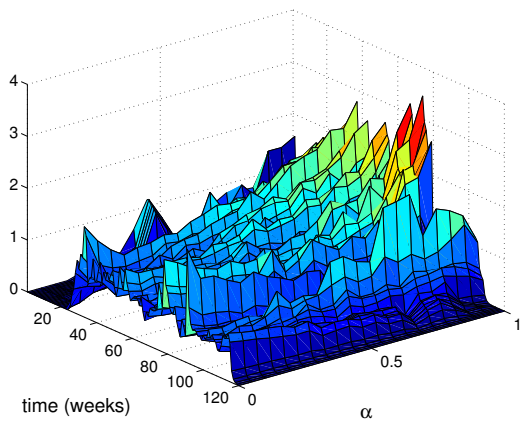

Fig. 6. Degree centrality for CEOs. Higher degree centrality for α near one signifies greater activity in the behavioral network. Anomalous behavior can be seen in the later time steps as activity patterns shift.

each other. The objective in this particular example is to show that using this method we can not only reduce noise, but also discover interesting multifaceted behavior that is not obvious from one layer alone.

Figure 5 shows the betweeness centrality of the Directors group over time as the mixing parameter is varied. In general, the centrality measure increases approximately monotonically as α is varied; however, from week 95 to week 115, betweenness centrality is significantly increased when using a combined dynamic network—that is, an intermediate value of α. This time corresponds to the beginning of the company's upheaval and public disclosure of troubles. Perhaps by examining both network layers simultaneously we have removed some of the edges between other classes, and thus the centrality score of this particular group increased. It is true that during this time, when overall email usage increased, the betweenness centrality measure went down, as there were more shortest paths through users from other groups. Using the combination of layers, however, there appears to be an increase in the number of shortest paths through the Directors group.

On the other hand, we can also see well-behaved monotonic correlations in some cases. Figure 6 shows a transition of degree centrality for the class of CEOs (of which there were four during this time period). The behavioral network shows more connectivity for the CEO class. This phenomenon makes sense, as the behavioral data takes into account all written documents, which could be correlated with those of other users, while the relational network only takes into account direct communication between the CEOs and others. In reality, much of that communication is performed through third parties (such as assistants), and thus CEOs probably do not send as much email as the average employee. Increasingly anomalous behavior occurs toward the end of the time period. We hypothesize that this is due to a larger volume of unusual emails sent directly to the CEO during this tumultuous period.

VI. Conclusion

We introduced a novel method for inference on multilayer networks. A hierarchical model was used to jointly describe the noisy observation matrices and MAP estimation was performed on the relevant latent variable. A simulation example using clustering demonstrated that the mixture of layers under the correct circumstances can lead to better results, and possibly a better understanding of the underlying structure between users. A real-data example was also discussed using the ENRON email dataset. This paper also leads the way for future work; in addition to trying more noise models that are not so simply reproduced or even non-convex, one can use multi-objective optimization to explore other objective functions that could be useful in describing a multi-layer network, such as network smoothness or the centrality distribution.

VII. Acknowledgements

We would like to thank Kevin Xu for providing the code for the DSBM model and his suggestions for utilizing it, as well as his general comments on the content of the paper.

References

[1] A. Raftery, "Bayesian model selection in social research," *Sociological methodology*, vol. 25, pp. 111–164, 1995.

[2] M. Ehrgott, "Multiobjective optimization," *AI Magazine*, vol. 29, no. 4, pp. 47–57, Winter 2008. [Online]. Available: http://search.proquest.com.proxy.lib.umich.edu/docview/208128027?accountid=14667

[3] X.-S. Yang, *Multiobjective Optimization*. John Wiley and Sons, Inc., 2010, pp. 231–246. [Online]. Available: http://dx.doi.org/10.1002/9780470640425.ch18

[4] P. Ngatchou, A. Zarei, and M. El-Sharkawi, "Pareto multi objective optimization," in *Intelligent Systems Application to Power Systems, 2005. Proceedings of the 13th International Conference on*, 2005, pp. 84–91.

[5] A. M. Geoffrion, "Proper efficiency and the theory of vector maximization," *Journal of Mathematical Analysis and Applications*, vol. 22, no. 3, pp. 618 – 630, 1968. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0022247X68902011

[6] K. S. Xu and A. O. H. III, "Dynamic stochastic blockmodels: Statistical models for time-evolving networks," *CoRR*, vol. abs/1304.5974, 2013.

[7] L. Hubert and P. Arabie, "Comparing partitions," *Journal of Classification*, vol. 2, no. 1, pp. 193–218, 1985. [Online]. Available: http://dx.doi.org/10.1007/BF01908075

[8] M. Baena-Garcia, J. Carmona-Cejudo, G. Castillo, and R. Morales-Bueno, "Tf-sidf: Term frequency, sketched inverse document frequency," in *Intelligent Systems Design and Applications (ISDA), 2011 11th International Conference on*, 2011, pp. 1044–1049.