Probabilistic Community and Role Model for Social Networks

Yu Han[†] and Jie Tang[†‡♯]
[†]Department of Computer Science and Technology, Tsinghua University
[‡]Tsinghua National Laboratory for Information Science and Technology (TNList)
[♯]Jiangsu Collaborative Innovation Center for Language Ability, Jiangsu Normal University, China
yuhanthu@126.com, jietang@tsinghua.edu.cn

ABSTRACT

Numerous models have been proposed for modeling social networks to explore their structure or to address application problems, such as community detection and behavior prediction. However, the results are still far from satisfactory. One of the biggest challenges is how to capture all the information of a social network such as links, communities, user attributes, roles and behaviors, in a unified manner.

In this paper, we propose a unified probabilistic framework, the Community Role Model (CRM), to model a social network. CRM incorporates all the information of nodes and edges that form a social network. We propose methods based on Gibbs sampling and an EM algorithm to estimate model parameters and fit our model to real social networks. Real data experiments show that CRM can be used not only to represent a social network, but also to handle various application problems with better performance than a baseline model, without any modification to the model.

Categories and Subject Descriptors

J.4 [**Social and Behavioral Sciences**]: Sociology; H.2.8 [**Database Applications**]: Data Mining

Keywords

Social Network, Community, Behavior Prediction

1. INTRODUCTION

Online social networks—e.g., Twitter, Facebook, Flickr—have become large complex virtual systems. Visible and invisible elements interact and affect each other. We can use a graph to model the structure of a social network, where nodes and edges represent users and interactions, which are visible elements. There are also dynamic visible elements—i.e., user actions, such as retweeting in Twitter and commenting in Flickr. Moreover, there are also invisible elements, such as *community* [13, 38] and *role* [46, 50], that affect the visible elements. Previous research, such as [10, 33, 45] and empirical studies on online social networks, including Facebook [44], Twitter [20], Flickr [31], YouTube [32], Yahoo!360 [18], Cyworld, Myspace and Orkut [1], revealed many interesting phenomena and basic underlying laws. For example, in a social network, nodes may have closer relationships within a community than across communities. Nodes may have different attributes—for example, some nodes may be popular, and have many followers, but others may be different. Nodes may exhibit different behaviors—for example, some nodes seem very active, and repost messages or comment on pictures, while others may not. However, according to [42], people's behaviors not only depend on their own attributes, but also on the influence of their neighbors and communities. How should we model a complex social network so that the model can capture the intrinsic relations between all these elements, such as conformity influence, individual attributes, and actions? How do we use a social network model to handle issues such as community detection and behavior prediction?

Social network analysis has been attracting much interest from researchers. Many models have been proposed to model the structure of a social network [15, 16, 22, 35, 36, 37, 48] and to handle issues such as social influence analysis [2, 9, 14, 28, 41, 42, 50], behavior prediction [39, 46], and link prediction [17, 23, 27, 40]. [15] uses latent space to model a social network in which every node is associated with a location in p-dimensional space, and two nodes are more likely to have links if they are closer. [48] describes a random graph model for social networks based on the dot product, which assigns each node a random vector, and quantifies the probability of a link between two nodes by the dot product of their vectors. [6] proposes a model that regards nodes as points in Euclidean space, and generates edges based on a mixture of the distances between nodes and a ranking function. [21] proposes a model to simulate the forming process of a social network with the Kronecker product of adjacency matrices. [16] introduces a multiplicative attribute graph model that uses the affinity of attributes of two nodes to indicate the potential for them to form a link. [46] takes the roles that nodes might play into consideration and proposes a model to predict information diffusion in a social network. [50] proposes a probabilistic model that combines the nodes' attributes and community influence to analyze nodes' behaviors. Although much progress has been made, the results of existing work are not satisfactory, due to their limitations:

1. Most social network models utilize only portions of the available social network information. For example, [42]

KDD'15, August 10-13, 2015, Sydney, NSW, Australia.

http://dx.doi.org/10.1145/2783258.2783274.

only takes link information into consideration, ignoring the differences between the nodes themselves, while [46] assumes three roles that nodes could play, ignoring the conformity influence in information diffusion.

2. Most models only focus on a few aspects of social networks, missing the global view. For example, some papers only focus on the static structure of social networks, while others focus only on user behaviors.

3. Many models are based on discriminative methods and have not capture the nature of social networks. As a result, they can only be used to settle specific issues. Such models may seems reasonable in some specific circumstances but not in others.

4. Some works use a deterministic method. However, this is usually impractical in complex social networks.

In this paper, we mine the intrinsic relationships between all visible and invisible elements of a social network, including communities, links, node attributes, roles and actions, and incorporate them into a unified probabilistic generative framework. The proposed model can also easily handle many practical issues in social networks, such as community detection and behavior prediction, without any modification to the model. To the best of our knowledge, this is the first model that captures all the information of a social network and can represent all its facets. The contributions of this paper include:

1. We incorporate various elements of a social network into a unified probabilistic generative framework, which can represent a complex social network better than other models. We further design a method to estimate the parameters of the model.

2. We use our model to generate a synthetic network with the learned parameters, and verify the superiority of our model to the baseline method in terms of six metrics.

3. We apply the model to two problems—behavior prediction and community detection—verifying its versatility and effectiveness.

This paper is organized as follows: In Section 2, we propose the Community Role Model (CRM) to model a social network and provide a method for parameter estimation and inference of CRM. We conduct two sets of experiments and a case study on real data sets in Section 3. Section 4 is a survey of related work. We conclude the paper in Section 5.

2. MODEL

2.1 Intuition

The intuition behind our model is that we can describe a social network as follows:

First, a social network is composed of many nodes/users, and each node is associated with many edges/links. [12] offers an edge-distribution law, stating that the distribution of edges is usually locally inhomogeneous, and highly concentrated within special groups of nodes, but sparse between these groups. In other words, each node may belong to several communities, and whether it has a link to other nodes might depend on the communities to which it belongs. Thus we can assume that each node has a distribution over the communities—i.e., that different nodes may be located in different communities. A node in a specific community may have a unique probability to link to another node. For example, as Figure 1 shows, in community c_1, v has a higher probability to link to u and a lower probability to link to w. However, the situation is reversed when v belongs to community c_2.

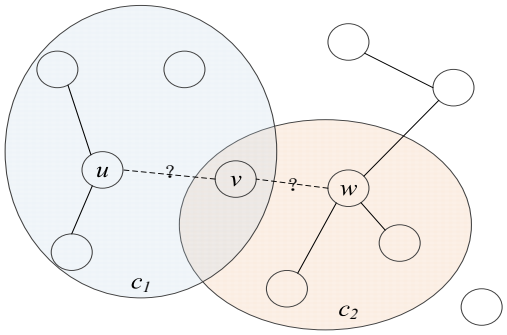


Figure 1: Social network

Second, each node has many attributes, such as in-degree, out-degree, and other attributes. Based on these attributes, we can classify the nodes into clusters. Each cluster can be regarded as a role that nodes play. For example, some nodes may have higher in-degree, and play the role of *opinion leader* [30], while others may have higher out-degree, and tend to transfer messages across communities, playing the role of *structural hole spanner* [29]. The attributes of each role satisfy a specific distribution—such as a Gaussian distribution. Each node has a distribution over roles according to its attributes.

Last, each node may take some actions, such as transferring a message, commenting on other people's pictures, or following others. Most nodes tend to take similar actions with nodes in the same community; in other words, whether a node takes a specific action partly depends on the community it belongs to. Moreover, whether a node takes an action may also depend on the role it plays. For example, according to [29], 25% of information diffusion is controlled by 1% of nodes serving the role of structural hole spanners. Thus, when we predict the action that a node might take, we must consider the distributions that the node has over both communities and roles.

2.2 Formulation

We use $G = (V, E, X)$ to denote the structure of a social network, where V is the set of all users and E is an $N \times N$ matrix, with each element $e_{v,u} = 0$ or 1 indicating whether user v has a link to/with user u. We use the cardinality $|V| = N$ to denote the number of the users. The set of edges that associate with v is denoted as E_v. Notation X denotes an attributes matrix with size $N \times H$, where H is the number of all attributes. Each element $x_h^{(v)} \in X$ denotes the h-th attribute of user v. Unlike the value of e, $x_h^{(v)}$ is continuous.

Definition 1. **Community.** A social network consists of multiple communities, denoted as $c = [1, 2, ..., C]$. Each community has a multinomial distribution over all pairs

Table 1: Notations in the CRM model

SYMBOL	DESCRIPTION
C	number of communities
R	number of roles
$e_{v,u}$	the edge between v and u
$x_h^{(v)}$	the h-th attribute of node v
$y_m^{(v)}$	the m-th action of node v
$z_{v,i}$	the community that the i-th edge of node v is assigned to
d_v	the role that node v is assigned to
$\phi^{(v)}$	multinomial distribution over communities specific to node v
$\theta^{(v)}$	multinomial distribution over roles specific to node v
$\zeta^{(c)}$	multinomial distribution over edges/nodes specific to community c
$\rho^{\tau,r}$	multinomial distribution over actions specific to community-role pair (τ, r)
$\mu_{r,h}$	mean of h-th attribute specific to role r
$\sigma_{r,h}$	standard deviation of h-th attribute specific to role r

(v, u), denoted as ζ. For a directed graph, the edge $e_{v,u}$ and $e_{u,v}$ share one item in the parameters—i.e., the pair (v, u), in community distributions. $\zeta_{v,u}^{(c)}$ denotes the probability of $e_{v,u}/e_{u,v}$ in community c, subject to $\sum_{v,u} \zeta_{v,u}^{(c)} = 1$. Note that, since edges are denoted by nodes, we could easily transform the distribution of communities over edges into distribution over nodes, which conforms to the usual definition of community and become easier to understand in some circumstances, such as community detection.

Definition 2. **Node Distribution over Communities.** Each node has a multinomial distribution over communities, which is denoted as ϕ. $\phi_c^{(v)}$ denotes the probability for v to be located in c, and is subject to $\sum_c \phi_c^{(v)} = 1$.

Definition 3. **Role.** A node may play multiple different roles, denoted as $r = [1, 2, ..., R]$. Each role has a set of parameters for the distribution the attributes conform to. Here we use Gaussian distribution. If a node plays role r, its h-th attribute conforms to $N(\mu_{r,h}, \sigma_{r,h}^2)$.

Definition 4. **Nodes Distribution over Roles.** Each node has a multinomial distribution over roles, which is denoted as θ. $\theta_r^{(v)}$ denotes the probability for v to play role r, and is subject to $\sum_r \theta_r^{(v)} = 1$.

Definition 5. **Action.** Each node can take some actions, such as transferring a message or following others. For different kinds of social networks, actions take different forms. Take the action of repost in a microblog network, for example. We use $y^{(v)}$ to denote a repost action of user v. We set time $t = 0$ as the start point. During time period $[0, T]$, there are M messages posted by the users that v follows. We use $y_m^{(v)} = 0$ or 1 $(i = 1, 2, ..., M)$ to denote whether v reposts the m-th message during a reasonable time period $[0, T']$.

Definition 6. **Community-Role Pair.** Whether a node would take an action depends on the communities it and its target belong to and the role it plays. We use ρ to denote the distribution of community-role pairs over actions. According to the above definition, action $y_m^{(v)}$ only contains two cases, so we can use a Bernoulli distribution to model the distribution of community-role pairs over actions. $\rho_m^{\tau,r}$ denotes the probability for $y_m = 1$, where $\tau = 1(c_v \neq c_u)$. $1(\cdot)$ is an indicator function. If $c_v \neq c_u$, $1(c_v \neq c_u) = 1$, otherwise 0. It is noted that the "community" in "community-role pair" represents whether the node and its target belong to the same community, so τ is binary, with 0 meaning the same community and 1 meaning different communities.

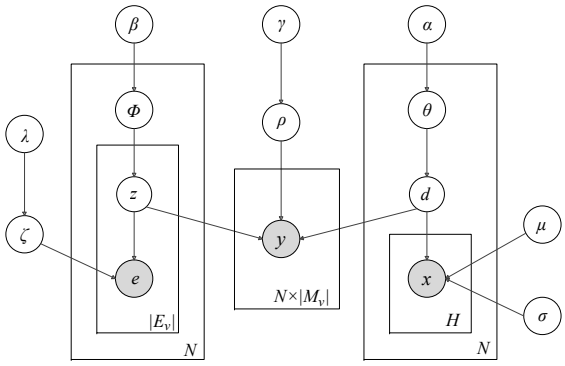


Figure 2: The CRM model

2.3 Model Description

Our goal is to devise a probabilistic generative model, CRM, to represent a social network by capturing relationships and interactions between all the elements of such a network, including links, node attributes, communities, roles, actions, etc. To do this, CRM assumes that a social network can be generated through three processes, with each process based on one of the three visible elements in a social network—edges, node attributes, and actions.

An edge is defined to be an item from a set indexed by $1, 2, \cdots, N^2$. (For an undirected graph, it is $1, 2, \cdots, N(N+1)/2$.) We represent edges using unit-basis vectors, of which only one component is 1 and all other components are 0. Each node is associated with a sequence of several edges denoted by $v = (e_{v,i}, e_{j,v})$, where $i \in I_{in}$ and $j \in I_{out}$. I_{in} is the set of tail endpoints adjacent to v, while I_{out} is the set of head endpoints adjacent to v. For an undirected graph, each edge of v can be denoted with the node which is the edge's endpoint adjacent to v. Each node belongs to several communities. Thus we can regard a node as a random mixture over communities. The generative process of all edges in a social network can be described as follows:

For each node v in the graph:

1. Draw ζ from Dirichlet(λ);
2. Draw a ϕ_v from Dirichlet(β) prior;
3. For each edge $e_{v,i}$:
   - Draw a community $z_{v,i} = c$ from multinomial distribution ϕ_v;
   - Draw an edge $e_{v,i}$ from a multinomial $\zeta^{(c)}$ specific to community c.

The time complexity of the above process is $O(nonezeros(E))$, where $nonezeros(E)$ denotes the number of nonzero items in E. The distribution of edges E is as:

$$p(E|\beta,\lambda) = \int p(\zeta|\lambda)\prod_{v}\int p(\phi_v|\beta) \cdot \prod_{|E_v|}\sum_{z_{v,i}} p(z_{v,i}|\phi_v)p(e_v|z_{v,i},\zeta)d\phi_v d\zeta. \quad (1)$$

Each node plays several roles and is associated with a sequence of several attributes, denoted by $v = (x_h)$, where $h = [1, 2, ..., H]$. We define each role as a distribution over attributes and each node is a random mixture over roles. The generative process of all nodes in a social network can be described as follows:

For each node v in the graph:

1. Draw a θ_v from Dirichlet(α) prior;
2. Draw a role $d_v = r$ from multinomial distribution θ_v;
3. For each attribute of v, draw a value $x_h^{(r)} \sim G(\mu_{r,h}, \sigma_{r,h}^2)$.

The time complexity of the above process is $O(NH)$. The joint distribution of attributes X is defined as:

$$p(X|\alpha,\mu,\sigma) = \prod_{v}\int p(\theta_v|\alpha) \cdot \sum_{d_v} p(d_v|\theta_v)\prod_{h} p(x_h^{(v)}|d_v,\mu_{r,k},\sigma_{r,k})d\theta_v. \quad (2)$$

Regarding actions, each node is associated with a sequence of several actions denoted by $v = (y_m)$, where $m = [1, 2, ..., M]$. The generative process of the actions can be described as follows:

For each action y_m:

1. Draw ρ from Dirichlet(γ) prior;
2. Draw a community c_v for v from ϕ_v;
3. Draw a community c_u for u, which post the message m, from ϕ_u;
4. Draw a role r from θ_v;
5. Draw $y_m \sim Bernoulli(\rho^{\tau,r})$.

The time complexity of the above process is $O(NM)$. The joint distribution of actions Y is defined as:

$$p(Y|\gamma,\phi,\theta) = \int p(\rho_{\tau,r}) \prod_{v}\sum_{\tau}\sum_{r} p(r|\theta_v)p(\tau|\phi_v)p(y_m^{(v)}|\rho_{\tau,r})d\rho_{\tau,r}. \quad (3)$$

2.4 Inference and Parameters Estimation

It is intractable to directly solve the above distribution functions. We use Gibbs sampling to estimate ϕ and ζ.

The posterior probability of $z_{v,i}$ is calculated by

$$p(z_{v,i} = c|\mathbf{z}_{-v,-i}, E) \propto \frac{n_{-v,-i,c}^{(v)} + \beta}{|E_v| + |C|\beta}\frac{n_{-v,-i,c}^{(e)} + \lambda}{n_{-v,-i,\cdot}^{(e)} + |E|\lambda}. \quad (4)$$

After Gibbs sampling, parameters ϕ and ζ can be estimated by:

$$\phi_{v,c} = \frac{n_{v,c} + \beta}{|E_v| + |C|\beta}, \quad (5)$$

$$\zeta_{c,e} = \frac{n_{c,e} + \lambda}{n_c + |E|\lambda}. \quad (6)$$

We use an EM algorithm to iteratively maximize the joint likelihood of users' attributes X and to estimate parameters θ and η. The likelihood of X can be written as:

$$\mathcal{L} = \prod_{v}\prod_{h}\sum_{d_v}\frac{\theta_{v,r}}{\sqrt{2\pi}\sigma_{r,h}} e^{-\frac{(x_{v,h}-\mu_{r,h})^2}{2\sigma_{r,h}^2}}. \quad (7)$$

In the E-step, we estimate the h-th item of θ given the current parameters by:

$$\theta_{v,r} = \frac{\prod_h (2\pi)^{-\frac{1}{2}}\sigma_{r,h}^{-1} e^{-\frac{(x_{v,h}-\mu_{r,h})^2}{2\sigma_{r,h}^2}}}{\sum_{d_v}\prod_h (2\pi)^{-\frac{1}{2}}\sigma_{r,h}^{-1} e^{-\frac{(x_{v,h}-\mu_{r,h})^2}{2\sigma_{r,h}^2}}}. \quad (8)$$

Then in the M-step, we update parameters μ and σ by the following equations. (Detailed derivation of θ, μ, and σ is given in Appendix.)

$$\mu_{r,h} = \frac{\sum_v \theta_{v,r}x_{v,h}}{\sum_v \theta v, r}, \quad (9)$$

$$\sigma_{r,h} = \sqrt{\frac{\sum_v \theta_{v,r}(x_{v,h} - \mu_{r,h})^2}{\sum_v \theta_{v,r}}}. \quad (10)$$

Because ϕ and θ have been estimated during the above processes, we only need to estimate ρ. Again, with Gibbs sampling, we first calculate the posterior probability of the (a_v, d_v) by the following equation:

$$p(a_v = \tau, d_v = r|a_{-v}, r_{-v}, \mathbf{y}) \propto (\phi_v\phi_v^T)\theta_v\frac{n_{-v,-m,\tau,r} + \gamma}{|M| + 2|H|\gamma}. \quad (11)$$

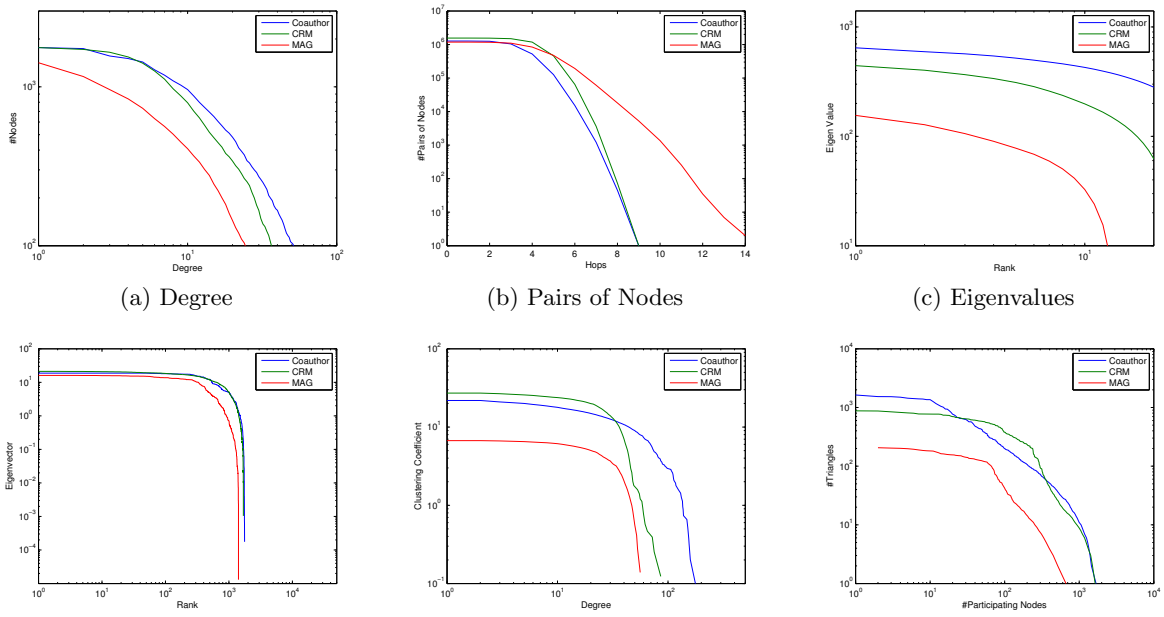
After sampling, the parameter ρ can be estimated by:

$$\rho = \frac{n_{v,m,\tau,r} + \gamma}{|M| + 2|H|\gamma}. \quad (12)$$

Model Applications. The learned CRM models can be used in various applications such as community discovery and behavior prediction. Essentially, parameters ζ represent (overlapping) communities discovered by CRM, while parameters ρ can be used to predict users' actions.

3. EXPERIMENTS

Now we evaluate the effectiveness of the proposed CRM model on real-world datasets. We first use a real dataset to learn the parameters of CRM. Then we use the parameters



(a) Degree (b) Pairs of Nodes (c) Eigenvalues

(d) Eigenvector (e) Clustering Coefficient (f) Triangle Participation Ratio

Figure 3: Metric values of the Coauthor network and the two networks generated by CRM and MAG. CRM outperforms MAG for every metric

to generate a synthetic social network that, ideally, should recover the original appearance. After that, we evaluate CRM by the following three tasks:

- **Structure recovery**. We compare the difference of structures between the generated synthetic network and the real network by means of six metrics: degree distribution, cluster coefficient, etc. Obviously, the more similar the features of the synthetic network and the real network, the better the model.
- **Behavior prediction**. CRM can predict users' actions by parameter ρ. We use four metrics, including precision, recall, F1-measure, and AUC, to evaluate the performance of CRM in predicting actions quantitatively.
- **Community detection**. CRM can mine communities by parameter ζ. We use a case study to demonstrate its effectiveness in detecting communities qualitatively.

3.1 Dataset

To evaluate CRM, we use three datasets.

The **Coauthor**[1] dataset is collected from [43], consisting of 1,712,433 computer science authors and 2,092,356 papers published by those authors between 1975 and 2012. For evaluation, we use a sub-network from [29], which contains 1765 authors, 13,415 corresponding collaboration relationships, and 7,233 papers published at 28 computer science conferences. These conferences can be divided into six fields: Artificial Intelligence(AI); Database(DB); Data Mining(DM); Distributed Parallel Computing(DP); Graphics, Vision and HCI (GV); Networks, Communication and Performance(NC). The conference list for each field can be found at [29]. We define an action of this network as publishing a paper in one of above research fields. Thus, there are six kinds of actions.

The **Facebook**[2] dataset is from [25], which contains information from 4,039 Facebook users and 88,234 links.

Weibo[3] is a popular microblogging service in China, which reports having more than 5 hundred million registered users. We use a sub-network from [49] with 1,776,950 users, 308,489,739 following relationships, 300,000 original messages and 23,755,810 repost actions.[4] All the messages were posted between Sep. 28th, 2012 and Oct. 29th, 2012. We classify all the original messages into ten topics, and define an action as posting or reposting a message in one topic, so the number of kinds of actions is ten.

3.2 Structure Recovery

We use the MAG model described by [16] as the baseline, which serves as a state-of-the-art method for modeling the structure of social networks. To demonstrate our model's superiority, we use the following network properties as our metrics to measure the difference of structure between the real network and the generated synthetic network. Part of the metrics are also used in [21] and [16], which represent the properties of a network from various aspects.

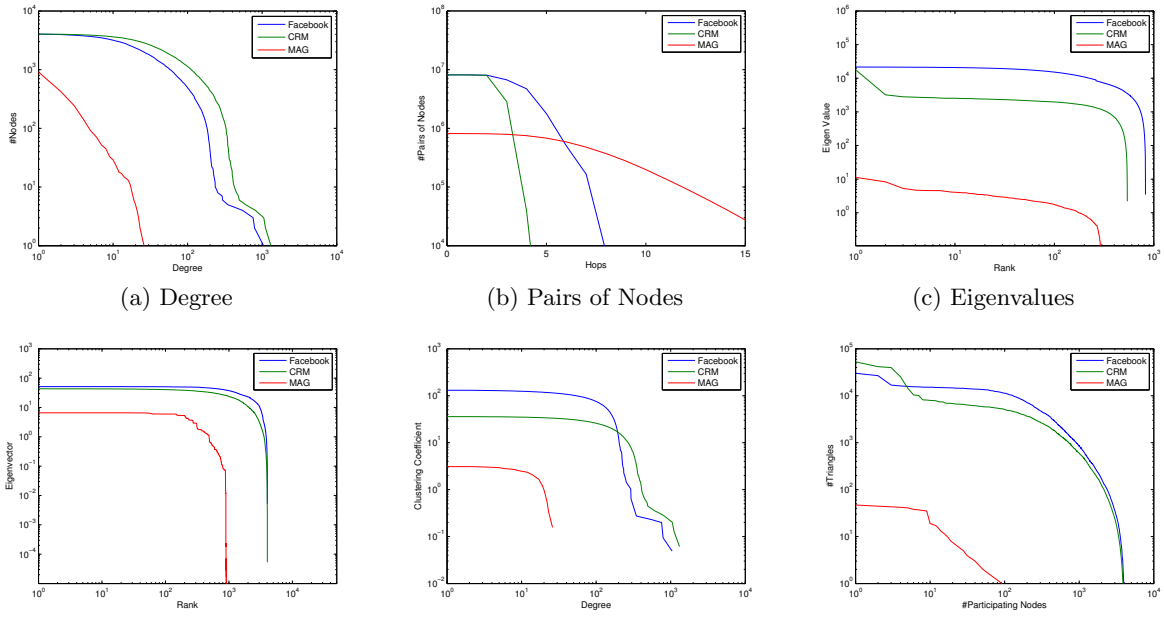
- **Degree** is the degree of nodes versus the number of corresponding nodes. As we know, it conforms to a power-law distribution in a scale-free network.
- **Pairs of Nodes** is the cumulative number of pairs of nodes that can be reached in $\leq h$ hops.

[1]https://aminer.org/billboard/AMinerNetwork

[2]http://www.facebook.com

[3]http://weibo.com

[4]https://aminer.org/billboard/Influencelocality



(a) Degree (b) Pairs of Nodes (c) Eigenvalues

(d) Eigenvector (e) Clustering Coefficient (f) Triangle Participation Ratio

Figure 4: Metric values of the Facebook network and the two networks generated by CRM and MAG. CRM outperforms MAG for every metric

- **Eigenvalues** are eigenvalues of the adjacency matrix representing the given network versus their rank.
- **Eigenvector** is the components of the leading eigenvector versus the rank.
- **Clustering coefficient** [45] is the average local clustering coefficient of nodes versus their degree.
- **Triangle Participation Ratio** is the number of triangles that a node is adjacent to versus the number of nodes.

We conduct this experiment on Coauthor and Facebook. For each dataset, we compute these values separately for the three networks: the real network, G; the generated network, G_{CRM} with our model; and the generated network G_{MAG} with the baseline model. Part of the code to compute the metric values is from [26]. Then we plot each metric of the three networks in one sub-figure in Figure 3 for Coauthor and Figure 4 for Facebook. Due to the heavy-tailed phenomenon of the metrics, we plot them in terms of cumulative distribution functions. Take the degree distribution, for example, the corresponding number of nodes for degree x is the number of nodes whose degrees are larger than x. From Figure 3 and Figure 4 we can see that both the networks generated by our model on the two datasets are more similar to the ground truth than to the baseline in all the above metrics, which signifies that our model is better than the baseline in modeling the structure of a social network.

3.3 Behavior Prediction

CRM can be also used to predict user behavior by parameter ρ. Given a social network G and action history A, we can build a training set $\{(\mathbf{x}_i, y_i)\}_{i=1,2,\cdots,n}$, where $\mathbf{x}_i$ is the attribute vector of a user and $y_i = a$ indicates that the user takes action a. Regarding baselines, we use existing classification algorithms, such as Support Vector Machine (SVM), Sequential Minimal Optimization (SMO), Logistic Regression (LR), Naive Bayes (NB), Gaussian Radial Basis Function Neural Network (RBF), and C4.5. We use Precision, Recall, F1-measure, and Area Under Curve (AUC) to evaluate the performance of each algorithm, and compare with the proposed CRM model.

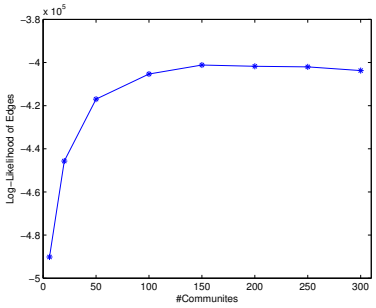
Table 3: Improvement shown by CRM over SVM, SMO, LR, NB, RBF, and C4.5 in terms of precision, recall, F1-measure, and AUC

Data Sets	Precision	Recall	F1-measure	AUC
Coauthor	0.37%	13.76%	7.04%	9.45%
Weibo	36.22%	40.14%	38.14%	32.08%

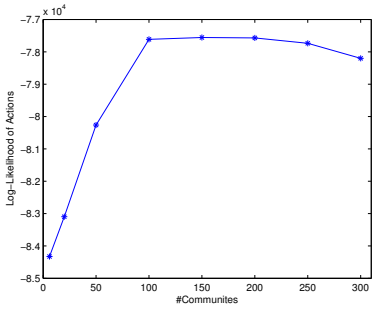
We conduct this experiment on Coauthor and Weibo. Table 2 lists the results of all comparison methods on the two datasets, and Table 3 gives the average improvement by CRM compared with the baseline methods. CRM clearly outperforms other methods on most metrics in Coauthor and Weibo. Take F1-measure, for example, in the Coauthor dataset, CRM results in a 7.04% improvement, and in Weibo, it achieves a 38.14% improvement on average. The improvement differences may lie in that whether a user posts or reposts a message has a stronger relation with his/her communities and friends, while whether a researcher publishes a paper in a specific area mostly depends on his/her attributes, having little to do with the influence of his/her communities and friends. CRM achieves much better performance in Weibo since it takes both communities and personal attributes into consideration, while other methods only take individual attributes into consideration. On the other hand, for researchers in Coauthor, CRM's superiority is less significant, because taking communities into consideration

Table 2: Average prediction performance of different methods on the Coauthor and Weibo datasets. The numbers enclosed in brackets are standard deviations.

Date set	Method	Precision	Recall	F1-measure	AUC
Coauthor	SVM	**0.8838(0.1725)**	0.5562(0.3183)	0.6827(0.2054)	0.7360(0.1111)
	SMO	0.8647(0.1218)	0.8142(0.1260)	0.8387(0.1138)	0.9218(0.0366)
	LR	0.8668(0.1242)	0.8292(0.1022)	0.8476(0.1016)	0.9642(0.0196)
	NB	0.8183(0.1830)	0.8115(0.1444)	0.8149(0.1549)	0.9417(0.0335)
	RBF	0.8552(0.1058)	0.8353(0.1165)	0.8451(0.1081)	0.9477(0.0271)
	C4.5	0.8328(0.0518)	0.8015(0.1286)	0.8169(0.1478)	0.9065(0.1165)
	CRM	0.8562(0.1490)	**0.8630(0.0598)**	**0.8596(0.1013)**	**0.9800(0.0199)**
Weibo	SVM	0.5067(0.1405)	0.5027(0.1185)	0.5047(0.1150)	0.6068(0.1113)
	SMO	0.5074(0.1464)	0.5209(0.1099)	0.5141(0.1271)	0.6145(0.0363)
	LR	0.5199(0.1306)	0.5469(0.1073)	0.5331(0.1157)	0.6330(0.0377)
	NB	0.5112(0.1245)	0.5692(0.1083)	0.5386(0.1172)	0.6397(0.0394)
	RBF	0.5225(0.1361)	0.4679(0.1117)	0.4937(0.1217)	0.5945(0.0085)
	C4.5	0.5237(0.1367)	0.5322(0.1114)	0.5279(0.1211)	0.6271(0.1083)
	CRM	**0.7017(0.1300)**	**0.7305(0.1079)**	**0.7158(0.1149)**	**0.8174(0.0233)**



(a) Sum of log-likelihood of edges changes with C



(b) Sum of log-likelihood of actions changes with C

Figure 5: The sums of log-likelihood of edges and actions change with C.

is less helpful in predicting whether a researcher publishes a paper in an area.

3.4 Case Study

CRM can be also used to detect communities with parameter ζ. For a new social network dataset, we must decide the number of communities C before detecting communities with CRM. To find the best C for the Coauthor dataset, we fix $R = 6$ [5] and set C =6, 20, 50, 100, 150, 200, 250, 300 sequentially, and compute the sums of log-likelihood for edges and actions with Eq.(13) and Eq.(14) separately serving as a posteriori measure to evaluate parameter C, and thus obtain Figure (5). The larger the sum of log-likelihood, the better the parameter. From Figure 5 we can easily believe that $C = 150$ may be the best choice for parameter C.

$$\mathcal{L}(edges) = \sum_{i=1}^{|E|} \ln p(e_i), \tag{13}$$

$$\mathcal{L}(actions) = \sum_{i=1}^{|Y|} \ln p(y_i). \tag{14}$$

Through the training of the model, we obtain the community distribution over nodes. We fix $C = 150$ and select three communities. Table 5 lists the representative five researchers with the highest probabilities in each community.

[5]We conducted an experiment verifying that the log-likelihood of actions in CRM on the Coauthor dataset is not very sensitive to the value of R, and $R = 6$ is slightly better than other values.

4. RELATED WORK

There are three types of research related to this work: network structure modeling, behavior prediction, and community detection.

Network structure modeling. Network structure modeling has a long history and has become a hot topic, attracting more and more interest from computer-science researchers. There is great interest in uncovering underlying principles with which networks comply. Early in 1960, [11] proposed a model that uses a real number $p \in (0, 1)$ to predict whether two nodes have a link between them, where p is determined by the scale of the network. [5] proposes a generative model, in which a graph is generated by adding nodes into an existing graph, and the probabilities of new nodes having links with existing nodes depend on the degree of existing nodes at that time. [8] proposes a model that constructs a sequence of nodes by some values. In that model, the probability that two nodes have a link is proportional to the product of the values of the two nodes. [19] proposes a model in which, when adding a node to an existing graph, selecting an existing node randomly and adding links with its neighbors with certain probabilities yields a model. [4] adopts a mechanism that not only adds but also deletes nodes when generating a network. [15] introduces

Table 4: Representative researchers in three different communities

Comm.	Name	Affiliation
1	Jiawei Han	UIUC
	Jian Pei	SFU
	Philip S. Yu	UIC
	Hong Cheng	CUHK
	Wei Wang	UNC
2	Thomas S. Huang	UIUC
	Yun Raymond Fu	UB
	Shuicheng Yan	NUS
	Mark A. Hasegawa-Johnson	UIUC
	Xiaoou Tang	CUHK
3	Philip A. Bernstein	Microsoft
	Nathan Andrew Goodman	UA
	David Dewitt	UW-Madison
	Erhard Rahm	U. of Leipzig
	Michael Stonebraker	MIT

latent space to model a social network, and [37] extends this concept into dynamic networks. However, all the above works ignore an important concept: community.

Behavior prediction. This work was first conducted by economists in the 1890s. Recently, many computer scientists have been working on this topic in the social network context. [7] conducts a famous experiment indicating that one's voting choices are susceptible to those of his/her friends. [42] predicts user behaviors with influence from his/her friends and communities. However, neither takes personal behavior patterns into consideration. [47] analyzes retweeting behavior in Twitter, and proposes a factor graph model to predict retweeting behavior. [51] leverages knowledge of user behavior in different networks to alleviate the data sparsity problem and enhance the predictive performance of user modeling. [3] analyzes click stream data and reveals key features of social network workloads, such as how frequently people connect to social networks and for how long, as well as the types and sequences of activities that users conduct on social networks. [46] studies the reposting actions of users in social networks. This paper classifies users into three roles—opinion leader, structural hole spanner, and ordinary user, which have different behavior patterns. Whether a user reposts a message greatly depends on the role it plays.

Community detection. Since the concept of community was raised formally, various methods to detect community have been proposed. Due to space limitations, we do not list them here. With the proliferation of community detection methods, evaluating them has become a hot topic. Modularity [34] is a kind of measure to evaluate community detection algorithms through comparing edge densities. [24] regards community quality as a function of its size, and offers a more-refined lens to examine community detection methods.

5. CONCLUSION

In this paper, we study how to model a social network, capturing all its information, such as links, communities, user roles, user attributes, and user actions. From the relationships between these objects, we devise a probabilistic generative framework, the Community Role Model, to define a social network model. We apply CRM to real-world datasets, and obtain better performance than that of a state-of-the-art baseline method. CRM can also be used to address various practical problems without any change to the model itself, showing its superiority.

Understanding the nature of social networks is very important for modeling them, and for addressing a series of problems attached to them. As for future work, it would be intriguing to mine more factors that affect network structure and user behaviors so as to simulate a dynamic social network. It is also interesting to integrate nonparametric methods into our model to base parameter value choices on the data itself.

Acknowledgements. We thank Myunghwan Kim for sharing the MAG code. The work is supported by the National High-tech R&D Program (No. 2014AA015103), National Basic Research Program of China (No. 2014CB340506, No. 2012CB316006), NSFC (No. 61222212), NSFC-ANR (No. 61261130588), National Social Science Foundation of China (No.13&ZD190), the Tsinghua University Initiative Scientific Research Program (20121088096), a research fund supported by Huawei Inc., and Beijing key lab of networked multimedia.

6. REFERENCES

[1] Y.-Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong. Analysis of topological characteristics of huge online social networking services. In *WWW'07*, pages 835–844, 2007.

[2] A. Anagnostopoulos, R. Kumar, and M. Mahdian. Influence and correlation in social networks. In *KDD'08*, pages 7–15, 2008.

[3] F. Benevenuto, T. Rodrigues, M. Cha, and V. Almeida. Characterizing user behavior in online social networks. In *SIGCOMM'09*, pages 49–62, 2009.

[4] B. Bollobás and O. Riordan. Robustness and vulnerability of scale-free random graphs. *Internet Mathematics*, 1(1):1–35, 2004.

[5] B. Bollobás, O. Riordan, J. Spencer, G. Tusnády, et al. The degree sequence of a scale-free random graph process. *Random Structures & Algorithms*, 18(3):279–290, 2001.

[6] A. Bonato, J. Janssen, and P. Prałat. A geometric model for on-line social networks. In *International Workshop on Modeling Social Media*, page 4, 2010.

[7] R. M. Bond, C. J. Fariss, J. J. Jones, A. D. Kramer, C. Marlow, J. E. Settle, and J. H. Fowler. A 61-million-person experiment in social influence and political mobilization. *Nature*, 489(7415):295–298, 2012.

[8] F. Chung and L. Lu. Connected components in random graphs with given expected degree sequences. *Annals of combinatorics*, 6(2):125–145, 2002.

[9] D. Crandall, D. Cosley, D. Huttenlocher, J. Kleinberg, and S. Suri. Feedback effects between similarity and social influence in online communities. In *KDD'08*, pages 160–168, 2008.

[10] D. Easley and J. Kleinberg. *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press, 2010.

[11] P. Erd6s and A. Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hungar. Acad. Sci*, 5:17–61, 1960.

[12] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3):75–174, 2010.

[13] M. Girvan and M. E. Newman. Community structure in social and biological networks. *PNAS*, 99(12):7821–7826, 2002.

[14] M. Gomez Rodriguez, J. Leskovec, and A. Krause. Inferring networks of diffusion and influence. In *KDD'10*, pages 1019–1028, 2010.

[15] P. D. Hoff, A. E. Raftery, and M. S. Handcock. Latent space approaches to social network analysis. *JASA*, 97(460):1090–1098, 2002.

[16] M. Kim and J. Leskovec. Modeling social networks with node attributes using the multiplicative attribute graph model. *arXiv preprint arXiv:1106.5053*, 2011.

[17] M. Kim and J. Leskovec. The network completion problem: Inferring missing nodes and edges in networks. In *SDM*, pages 47–58, 2011.

[18] R. Kumar, J. Novak, and A. Tomkins. Structure and evolution of online social networks. In *Link mining: models, algorithms, and applications*, pages 337–357. 2010.

[19] R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal. Stochastic models for the web graph. In *FOCS'00*, pages 57–65, 2000.

[20] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *WWW'10*, pages 591–600, 2010.

[21] J. Leskovec, D. Chakrabarti, J. Kleinberg, C. Faloutsos, and Z. Ghahramani. Kronecker graphs: An approach to modeling networks. *JMLR*, 11:985–1042, 2010.

[22] J. Leskovec and C. Faloutsos. Scalable modeling of real graphs using kronecker multiplication. In *ICML'07*, pages 497–504, 2007.

[23] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting positive and negative links in online social networks. In *WWW'10*, pages 641–650, 2010.

[24] J. Leskovec, K. J. Lang, and M. Mahoney. Empirical comparison of algorithms for network community detection. In *WWW'10*, pages 631–640, 2010.

[25] J. Leskovec and J. J. Mcauley. Learning to discover social circles in ego networks. In *NIPS'12*, pages 539–547, 2012.

[26] J. Leskovec and R. Sosič. Snap.py: SNAP for Python, a general purpose network analysis and graph mining tool in Python. `http://snap.stanford.edu/snappy`, June 2014.

[27] D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. *JASIST*, 58(7):1019–1031, 2007.

[28] L. Liu, J. Tang, J. Han, M. Jiang, and S. Yang. Mining topic-level influence in heterogeneous networks. In *CIKM'10*, pages 199–208, 2010.

[29] T. Lou and J. Tang. Mining structural hole spanners through information diffusion in social networks. In *WWW'13*, pages 825–836, 2013.

[30] R. K. Merton. *Social theory and social structure*. Simon and Schuster, 1968.

[31] A. Mislove, H. S. Koppula, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Growth of the flickr social network. In *The First workshop on Online social networks*, pages 25–30, 2008.

[32] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *SIGCOMM'07*, pages 29–42, 2007.

[33] M. Newman. *Networks: an introduction*. Oxford University Press, 2010.

[34] M. E. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113, 2004.

[35] G. Palla, L. Lovász, and T. Vicsek. Multifractal network generator. *PNAS*, 107(17):7640–7645, 2010.

[36] G. Robins, P. Pattison, Y. Kalish, and D. Lusher. An introduction to exponential random graph (p*) models for social networks. *Social networks*, 29(2):173–191, 2007.

[37] P. Sarkar and A. W. Moore. Dynamic social network analysis using latent space models. *ACM SIGKDD Explorations Newsletter*, 7(2):31–40, 2005.

[38] J. Scott. *Social network analysis*. Sage, 2012.

[39] C. Tan, J. Tang, J. Sun, Q. Lin, and F. Wang. Social action tracking via noise tolerant time-varying factor graphs. In *KDD'10*, pages 1049–1058, 2010.

[40] J. Tang, T. Lou, and J. Kleinberg. Inferring social ties across heterogeneous networks. In *WSDM'12*, pages 743–752, 2012.

[41] J. Tang, J. Sun, C. Wang, and Z. Yang. Social influence analysis in large-scale networks. In *KDD'09*, pages 807–816, 2009.

[42] J. Tang, S. Wu, and J. Sun. Confluence: Conformity influence in large social networks. In *KDD'13*, pages 347–355, 2013.

[43] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su. Arnetminer: extraction and mining of academic social networks. In *KDD'08*, pages 990–998, 2008.

[44] J. Ugander, B. Karrer, L. Backstrom, and C. Marlow. The anatomy of the facebook social graph. *arXiv preprint arXiv:1111.4503*, 2011.

[45] D. J. Watts and S. H. Strogatz. Collective dynamics of ąősmall-worldąŕnetworks. *Nature*, 393(6684):440–442, 1998.

[46] Y. Yang, J. Tang, C. W.-k. Leung, Y. Sun, Q. Chen, J. Li, and Q. Yang. Rain: Social role-aware information diffusion. 2015.

[47] Z. Yang, J. Guo, K. Cai, J. Tang, J. Li, L. Zhang, and Z. Su. Understanding retweeting behaviors in social networks. In *CIKM'10*, pages 1633–1636, 2010.

[48] S. J. Young and E. R. Scheinerman. Random dot product graph models for social networks. In *Algorithms and models for the web-graph*, pages 138–149. 2007.

[49] J. Zhang, B. Liu, J. Tang, T. Chen, and J. Li. Social influence locality for modeling retweeting behaviors. In *IJCAI'13*, pages 2761–2767, 2013.

[50] J. Zhang, J. Tang, H. Zhuang, C. W.-K. Leung, and J. Li. Role-aware conformity influence modeling and analysis in social networks. In *AAAI'14*, pages 958–965, 2014.

[51] E. Zhong, W. Fan, J. Wang, L. Xiao, and Y. Li. Comsoc: adaptive transfer of user behaviors over composite social network. In *KDD'12*, pages 696–704, 2012.

APPENDIX

A. ESTIMATING θ, μ AND σ

From Eq.(7), we get the log-likelihood of attributes of all nodes, as in Eq.(15).

$$\begin{aligned}
\mathcal{L} &= \sum_v \sum_h \ln p(x_h^v; \theta, \mu, \sigma) \\
&= \sum_v \sum_h \ln \sum_{d_h^{(v)}=1}^{R} p(x_h^v | d_h^{(v)}; \mu, \sigma) p(d_h^{(v)}; \theta) \\
&= \sum_v \sum_h \ln \sum_{r=1}^{R} Q(d_h^{(v)} = r) \frac{p(x_h^v | d_h^{(v)}; \mu, \sigma) p(d_h^{(v)}; \theta)}{Q(d_h^{(v)} = r)} \\
&\geq \sum_v \sum_h \sum_{r=1}^{R} Q(d_h^{(v)} = r) \ln \frac{p(x_h^v | d_h^{(v)}; \mu, \sigma) p(d_h^{(v)}; \theta)}{Q(d_h^{(v)} = r)} \\
&= \sum_v \sum_h \sum_{r=1}^{R} w_r^v \ln \frac{p(x_h^v | d_h^{(v)}; \mu, \sigma) p(d_h^{(v)}; \theta)}{w_r^v} \\
&= \sum_v \sum_h \sum_{r=1}^{R} w_r^v \ln \frac{\frac{\theta_{v,r}}{\sqrt{2\pi}\sigma_{r,h}} e^{-\frac{(x_{v,h}-\mu_{r,h})^2}{2\sigma_{r,h}^2}}}{w_r^v}.
\end{aligned} \tag{15}$$

According to the Jensen Inequation, the inequality sign can be removed iff $\frac{p(x_h^v | d_h^{(v)}; \mu, \sigma) p(d_h^{(v)}; \theta)}{w_r^v} = c$, where c is a constant and w is a distribution d over r, so $\sum_r w_r^v = 1$. We can set

$$\begin{aligned}
w_r^v &= \frac{p(x_h^v, d_h^{(v)}; \theta, \mu, \sigma)}{\sum_r p(x_h^v, d_h^{(v)}; \theta, \mu, \sigma)} \\
&= \frac{\prod_h (2\pi)^{-\frac{1}{2}} \sigma_{r,h}^{-1} e^{-\frac{(x_{v,h}-\mu_{r,h})^2}{2\sigma_{r,h}^2}}}{\sum_r \prod_h (2\pi)^{-\frac{1}{2}} \sigma_{r,h}^{-1} e^{-\frac{(x_{v,h}-\mu_{r,h})^2}{2\sigma_{r,h}^2}}}.
\end{aligned} \tag{16}$$

First we assume that we have the values of μ and σ, then we maximize the lower bound of $\mathcal{L}$ by updating θ. Since $\sum_r \theta_{v,r} = 1$, we get Eq.(17) through the Lagrange Multiplier.

$$\mathcal{L}_\theta = \sum_v \sum_h \sum_{r=1}^{R} w_r^v \ln \frac{\frac{\theta_{v,r}}{\sqrt{2\pi}\sigma_{r,h}} e^{-\frac{(x_{v,h}-\mu_{r,h})^2}{2\sigma_{r,h}^2}}}{w_r^v} - \epsilon(\sum_r \theta_{v,r} - 1). \tag{17}$$

We compute the derivative of Eq.(17) with regard to $\theta_{v,r}$, and obtain Eq.(18).

$$\frac{\partial \mathcal{L}_\theta}{\partial \theta_{v,r}} = \sum_h \frac{w_r^v}{\theta_{v,r}} + \epsilon. \tag{18}$$

We set Eq.(18) to 0, and get $\frac{w_r^v}{\theta_{v,r}} = \frac{\epsilon}{H} = constant$. Because $\sum_r \theta_{v,r} = \sum_r w_r^v = 1$, we get Eq.(19).

$$\theta_{v,r} = w_r^v. \tag{19}$$

Then we maximize the lower bound of $\mathcal{L}$ by computing its derivative with regard to μ.

$$\begin{aligned}
&\nabla_{\mu_{r,h}} \sum_v \sum_h \sum_{r=1}^{R} w_r^v \ln \frac{\frac{\theta_{v,r}}{\sqrt{2\pi}\sigma_{r,h}} e^{-\frac{(x_{v,h}-\mu_{r,h})^2}{2\sigma_{r,h}^2}}}{w_r^v} \\
&= -\nabla_{\mu_{r,h}} \sum_v w_r^v \frac{(x_h^{(v)} - \mu_{r,h})^2}{2\sigma_{r,h}^2} \\
&= \sum_v w_r^v \frac{x_h^{(v)} - \mu_{r,h}}{\sigma_{r,h}^2}.
\end{aligned} \tag{20}$$

We set Eq.(20) to 0, and get Eq.(21).

$$\mu_{r,h} = \frac{\sum_v w_r^v x_h^{(v)}}{\sum_v w_r^v}. \tag{21}$$

Next we maximize the lower bound of $\mathcal{L}$ by computing its derivative with regard to σ.

$$\begin{aligned}
&\nabla_{\sigma_{r,h}} \sum_v \sum_h \sum_{r=1}^{R} w_r^v \ln \frac{\frac{\theta_{v,r}}{\sqrt{2\pi}\sigma_{r,h}} e^{-\frac{(x_{v,h}-\mu_{r,h})^2}{2\sigma_{r,h}^2}}}{w_r^v} \\
&= \nabla_{\sigma_{r,h}} \sum_v w_r^v (\ln \sigma_{r,h} - \frac{(x_h^{(v)} - \mu_{r,h})^2}{2\sigma_{r,h}^2}) \\
&= \sum_v -w_r^v (\frac{(x_h^{(v)} - \mu_{r,h})^2}{2\sigma_{r,h}^2} - \sigma_{r,h}^{-1}).
\end{aligned} \tag{22}$$

We set Eq.(22) to 0, and get Eq.(23).

$$\sigma_{r,h} = \sqrt{\frac{\sum_v w_r^v (x_h^{(v)} - \mu_{r,h})^2}{\sum_v w_r^v}}. \tag{23}$$

Combining Eq.(16), Eq.(19), Eq.(21) and Eq.(23), we get Eq.(8), Eq.(9) and Eq.(10).