Mixed-Membership Stochastic Block-Models for Transactional Networks

Bihan Zhuang, Lin Zuo

May 1, 2018

1 Literature Review

1.1 Background

1.1.1 Transactional Data and Data Representation

Most models for social network data assume that the relationship occurs between pairs of nodes and the relationship usually takes binary values. However, the assumptions do not always hold. In many cases, the data are transactional, with multiple nodes involved in each interaction. In the context of this paper, transactional data contains a list of one-to-many communications among nodes in a social network. One typical example is email data, in which for each email communication, the sender can send to one or multiple recipients at the same time. Additional possible information that goes along with transactional data includes timestamps, text of the message, and recipient classes.

In this paper, authors proposed a hierarchical Bayesian block-model that incorporates the transactional nature of data. The major motivation behind their work is to fully take advantage of co-recipient information and count information that are embedded in the raw transactional data. Previous methods for transactional data modeling ignore and waste these information. Because email data is a representative application, we use that as the example to explain data representation. The graph below (see Figure 1) demonstrates the conventional approach to represent transactional data. The left graph is raw transactional data where each row represents one email. Based on the raw transactional data, we count the number of interactions between pairs of nodes and transform the raw data to a matrix that contains transaction counts for each pair of nodes, shown in the middle graph. We assume that no node can be the sender and the recipient at the same time for any email. During the transformation from the left graph to the middle graph, information about co-recipient of the same message is lost. We then set the threshold for counts as 1 and convert any value larger than or equal to 1 to 1 and others to 0. Binary relations obtained by thresholding counts at 1 are represented by the socio-matrix in the right graph. During the transformation from the middle graph to the right graph, information about frequency

(a) Transactions

Sender	A	B	C	D
A	.	1	0	0
A	.	1	0	1
C	0	1	.	0
B	1	.	0	1
C	0	1	.	0

(b) Transaction counts

	(recipients)			
(sender)	A	B	C	D
A	.	2	0	1
B	1	.	0	1
C	0	2	.	0
D	0	0	0	.

(c) Binary relations

	(recipients)			
(sender)	A	B	C	D
A	.	1	0	1
B	1	.	0	1
C	0	1	.	0
D	0	0	0	.

Figure 1: *An example to demonstrate traditional representation of transactional data. (a) Original raw transaction data (b) Matrix of transaction counts (c) Socio-matrix after thresholding the count numbers.*

of interactions between nodes is lost. Under this approach, directional information among all nodes are well preserved, but information about whether two people receive the same email together and the number of interactions between pairs of nodes are ignored.

1.1.2 Related Research

The foundational work for this paper is Mixed Membership Stochastic Block-model (MMSB)[2], which seeks to model socio-matrices and binary relations. It incorporates mixed membership behaviors by allowing each node to be with a different group every time we sample a relation for this node. However, direct application of the MMSB model to transactional data involves the data simplification process explained in the last section, which discards co-recipient information and message frequency information. The new model proposed in this paper will solve this issue.

Kahanda and Neville's work on transactional data also provides insights on the importance of preserving transactional feature of the network in modeling[4]. In their paper, they adopted a supervised learning approach to predict link strength between nodes from transactional information and showed that network-transactional features had the largest impact on model performances. In addition to that, Kurihara, Kameya and Sato's work on frequency-based infinite relational model shows that adding the frequency of relation into infinite relational model improves clustering accuracy and thus suggests that counts of interactions is an important feature[5].

1.2 Transactional Mixed Membership Block Model

1.2.1 Model Setup

The transactional mixed membership block model (TMMSB) is explained here using the language of an email network. Assume N messages are sent within a network of M nodes. Each message n has a sender S_n and S_n itself can't be a recipient. Each message n also has a recipient list represented by M binary variables $Y_{n1}, ..., Y_{nM}$, where $Y_{nm} = 1$ when node m receives message n from S_n and $Y_{nm} = 0$ when node m does not receive message n from S_n. Note that if $S_n = i$, then $Y_{ni} = 0$, since a sender does not send to itself.

Suppose there are K groups in the network, we define a $K \times K$ interaction matrix B in which entry B_{kl} represents the probability of a node i in group k sending a message to a node j in group l. Entries of B are restricted to be between 0 and 1. Based on this basic block model setting, we further assume that each node can have different group memberships instead of belonging to just one group. In other words, a node i would have independent membership sampled for each message. This mixed membership part of the model is incorporated by adding additional hierarchy into the model – for each node i define a K-dimensional membership probability π_i, with $\sum_{k=1}^{K} \pi_{ik} = 1$.

Under this model setup, the observed variables are Y as described above and senders S_n. The interaction matrix B and group membership probabilities $\pi_1, ..., \pi_M$ must all be estimated.

1.2.2 Generative Process

The generative process for transactional mixed membership stochastic block model is outlined in **Algorithm 1**. The goal is to generate senders for each email as well as the Y matrix. Each node has a mixed membership vector π_i drawn from a Dirichlet distribution with hyperparameter α. For each email n, each node i will have its group assignment vector z_{ni} sampled from its membership vector π_i. Define z_{ni} as a binary K-dimensional vector with exactly one nonzero element, at the index of the group it belongs to. The sender S_n of each email is selected based on the sender probability λ. The recipients are sampled as $M-1$ Bernoulli random variables, with probability $z_{nS_n} B z_{nj}^T$, which indicates the selection of the entry in the ground truth B corresponding to the current group membership of the sending node S_n to the (potential) receiving node j. In other words, this is the probability that sender's group sends an email to the (potential) recipient's group. The resulting Y is a $N \times M$ matrix in which every row is a transaction and 1's in each row represent the recipients of the corresponding email.

The joint distribution over all latent variables and observed variables according to distributions specified in this section is then

$$p(Y, S, \pi_{1:M}, \lambda_{1:M}, Z_{1:N,1:M} | \alpha, B, \mu) = \prod_{m=1}^{M} p(\pi_m | \alpha) \prod_{m=1}^{M} p(\lambda_m | \mu) \times$$

$$\prod_{n=1}^{N} \Big[\prod_{m=1}^{M} p(z_{nm} | \pi_m) \times \prod_{m=1, m \neq S_n}^{M} p(Y_{nm} | Z_{nm}, Z_{nS_n}, B) \Big]$$

Algorithm 1 Generative Process for TMMSB

$M \leftarrow$ number of nodes
$N \leftarrow$ number of emails
$K \leftarrow$ number of groups
$\alpha \leftarrow$ parameter for Dirichlet distribution
$B \leftarrow$ ground truth interaction matrix
for $i \leftarrow 1$ to M **do**
  Draw mixed membership vector $\pi_i \sim Dirichlet(\alpha)$
  Draw sender probability $\lambda_i = \frac{exp(\lambda_i)}{\sum_{k=1}^{M} exp(\lambda_k)}$
end for
Choose $N \sim Possion(\epsilon)$ number of emails
for $n \leftarrow 1$ to N **do**
  Draw $z_{ni} \sim Multinomial(\pi_i)$
  Pick node u as sender with probability λ_u
  for $j \leftarrow 1$ to N **do**
    if $j \neq u$ **then**
      Draw $Y_{nj} \sim Bernoulli(z_{nu} B z_{nj}^T)$
    end if
  end for
end for

1.2.3 Inference

Our focus is to estimate groups and membership and ultimately recover the B matrix. So in the following sections we condition on senders S_n, eliminating the need to infer the λ's.

Due to the form of the joint distribution written in the previous section, the posterior in this model involves a multidimensional integral and summations, making it intractable. Instead, we use variational inference methods to approximate the true posterior and recover empirical estimates for the B interaction matrix. We pick a variational distribution over latent variables with free parameters. This distribution approximates the true posterior in terms of Kullback-Leibler divergence by optimizing the free parameters. A fully-factorized mean-field family of distributions is used in our variational distribution:

$$q(\pi_{1:M}, Z_{1:N,1:M}) = \prod_{m=1}^{M} q_1(\pi_m|\gamma_m) \prod_{n=1}^{N} \prod_{m=1}^{M} q_2(z_{nm}|\phi_{nm}) \tag{1}$$

where q_1 is a Dirichlet and q_2 is a Multinomial distribution. The set of free variational parameters $\{\gamma_{1:M}, \phi_{1:N,1:M}\}$ will be optimized to tighten the bound between the true posterior and the variational distribution.

We use the following equations for updating variational parameters. The update for ϕ_{nm} is

$$\begin{aligned} \phi_{nm,k} \propto\ & \exp(E_q(log(\pi_{m,k})))\times \\ & \prod_{l=1}^{K} \left(B_{lk}^{Y_{nm}}(1-B_{lk})^{1-Y_{nm}}\right)^{\phi_{nS_n,l}\times \mathbb{1}(m\neq S_n)} \times \\ & \prod_{m'\neq m} \prod_{l=1}^{K} \left(B_{kl}^{Y_{nm'}}(1-B_{kl})^{1-Y_{nm'}}\right)^{\phi_{nm',l}\times \mathbb{1}(m=S_n)} \end{aligned} \tag{2}$$

for all trasactions $n = 1, ..., N$ and all nodes $m = 1, ...M$. The update for γ_m is

$$\gamma_{m,k} = \alpha_k + \sum_{n=1}^{N} \phi_{nm,k} \tag{3}$$

for all nodes $m = 1, ...M$. The empirical Bayes estimate for B is

$$B_{k,l} = \frac{\sum_{n=1}^{N} \sum_{m=1,m\neq Sn}^{M} \phi_{nSn,k}\phi_{nm,l}Y_{nm}}{\sum_{n=1}^{N} \sum_{m=1,m\neq Sn}^{M} \phi_{nSn,k}\phi_{nm,l}} \tag{4}$$

The pseudocode for variational EM inference algorithm is presented in **Algorithm 2**. We optimize the free parameters in the E-step and optimize the B matrix in the M-step. The resulting B from this algorithm is the estimated B matrix.

Algorithm 2 Variational Bayesian Inference Algorithm

```
Initialize γ_mk = N/K for all m = 1,...M and k = 1,...K
Initialize φ_nmk = 1/K for all n = 1,...,N, m = 1,...M and k = 1,...K
Initialize B (will be specified later in Simulation Section)
Fix α = 0.1
repeat
  repeat
    for n ← 1 to N do
      for m ← 1 to M do
        for k ← 1 to K do
          Estimate φ_nmk with Eq. (2)
        end for
        Normalize φ_nmk to sum up to 1
      end for
    end for
    for m ← 1 to M do
      for k ← 1 to K do
        Estimate γ_m with Eq. (3)
      end for
    end for
  until convergence
  Estimate B with Eq. (4)
until convergence
{Convergence is reached when the log likelihood does not change anymore}
```

1.2.4 Model Choice

So far our inference algorithm is described under a fixed number of groups, K. To choose the correct number of groups, we employ a BIC criterion, which consists of a log-likelihood term and a penalty term. The log-likelihood term is composed of a sending term corresponding to the selection of the "sending" node for each transaction and a "receiving" term for choosing group memberships and recipients of this transaction. If we condition on the sender of an email, then the likelihood for recipient nodes is equivalent to $M-1$ Bernoulli trials (as described in the generative process). We can write the "receiving" term of the likelihood as

$$L = \prod_{n=1}^{N} \prod_{j \in 1:M, j \neq S_n} p_{ij}^{Y_{nj}} (1 - p_{ij})^{1-Y_{nj}} \tag{5}$$

where S_n is the sender node for transaction n, and $p_{ij} = \mathrm{P}(j \text{ receives } | i \text{ sends}) = \pi_i B \pi_j^T$. Note that since group assignments (z_{nm}'s) are unobserved, we use the average over group memberships in this calculation. Then, the BIC score for choosing the number of groups can be calculated using the following equation:

$$BIC = 2\mathrm{log}L - (K^2 + K) log(|Y|) \tag{6}$$

where $K^2 + K$ is the number of total parameters in the model (B and α) and $|Y| = \sum_{n,m} y_{nm}$ represents the number of total recipients in the network.

1.2.5 Clustering Performance Measures

In order to evaluate the clustering performance of the model and the accuracy of mixed membership probability vectors estimated for all nodes, the authors seek to obtain ground truth mixed membership vectors from data and compare estimated mixed membership vectors with true mixed membership vectors. The new clustering performance measures in this paper are developed based on Amig et al's extended BCubed metrics for overlapping clustering where an object can belong to more than one cluster[3]. Amig et al's measure consists of precision, recall and F-measure and is used for evaluating extrinsic clustering output. Precision is the fraction of data points that are assigned to the same cluster as they truly belong to. Recall is the fraction of data points from a true class that are assigned to the same cluster by the model. Different from the case discussed in Amig et al's work, in the context of TMMSB, instead of extrinsic clustering group, each node has a membership probability vector. For two nodes e and e', $\pi(\cdot)$ is the estimated membership probability vector and $\gamma(\cdot)$ is the true membership probability vector. The modified metrics are shown as follows:

$$Precision(e, e') = \frac{Min(\pi(e)^T \pi(e'), \gamma(e)^T \gamma(e'))}{\pi(e)^T \pi(e')} \tag{7}$$

$$Recall(e, e') = \frac{Min(\pi(e)^T \pi(e'), \gamma(e)^T \gamma(e'))}{\gamma(e)^T \gamma(e')} \tag{8}$$

We then get average precision and recall measures by averaging over all pairs of nodes. For the F-measure, we define it as the harmonic mean of precision and recall. This clustering performance measure will be later applied in the Reddit dataset analysis.

1.3 Data Analysis

The authors present empirical results on Enron email dataset and transactional data scraped from Reddit by using TMMSB.

1.3.1 Enron Dataset

The authors subset the original Enron email dataset by focusing on all messages sent in October and November, 2011, because of the high volume of communications happened within the company during that time period. They run the inference algorithm on the data and calculate BIC values for different numbers of groups K. Since BIC score is lowest when K=9, they decide to settle down on K=9. They then group employees by their most possible group (node i belongs to the group j where $\pi_{i,j}$ is the largest element in π_i) by using the estimated π's and B matrix from the inference algorithm. Following that, they calculate number of messages sent between pairs of groups by multiplying P(j receives | i sends) $= p_{ij} = \pi_i B \pi_j^T$ with the number of messages sent by person i. From there, they construct predicted message frequency matrix, with rows and columns ordered by group ids and compare the predicted matrix with the observed message frequency matrix.

The predicted message frequency matrix suggests that the model captures some important characteristics about inter-group relationship shown in the original message frequency matrix. For example, people in group 1 are more likely to send messages to people in group 3 instead of people in other groups. However, within each group, the predicted frequency matrix seems to be more homogenous than the original matrix, probably because sub-level behaviours are not considered by TMMSB. For example, although all nodes in group 9 almost exclusively send messages to other nodes in the same group, after further examining the interactions among all nodes in this group, the authors found that they could be further divided into two sub-blocks. Within each sub-block, nodes primarily communicate amongst themselves. TMMSB doesn't capture this sub-level behavior and assigns nodes from two sub-blocks to one single block because the model groups nodes by similar sending behavior, not by the sending of messages to the same individuals. In other words, although nodes in group 9 could be further categorized based on specific individuals that they send messages to, they still share the same sending behavior, which is only sending messages to members in the same group.

Besides all the results presented in the paper summarized above, we think there is one drawback about this analysis. Although the authors provide mathematical reasoning behind the clustering results, there is no direct interpretation about these nine group labels. Questions like, who are these people that are assigned to group 1 and why do people within group 9 only send messages to each other still remain unanswered. Original Enron dataset contains information about job titles for each individual. If information about job titles could be connected to these group labels in some way, the analysis results would provide more practical meanings. It is not mentioned in the paper whether there is any relationship between groups clustered by TMMSB and any type of real groups existed within the company.

1.3.2 Reddit Dataset

Reddit is an online social news platform where users can post links and comment on links posted by other people. Users can also follow up on comments and interact with users who post the comments. A list of comments for a comment is called a thread. For each post that users post, they can choose a topical section, "subreddits", to indicate which category that post falls under. In this analysis, the authors include both posts/comments and comments/follow-up comments as transactions. To be specific, for posts/comments transactions, users who post the link are considered senders and users who comment under the link are considered receivers. For comments/follow-up comments transactions, users who start the thread are considered senders and users who post follow-up comments are considered receivers. One big advantage of using Reddit data is that we can define a true mixed membership probability vector by referring to users' activities in all ten subreddits. For each user, the authors are able to get their frequency of posting in each subreddit and obtain the ground truth mixed membership probability vector by normalizing frequencies. In the paper, the authors scraped ten most active subreddits and only kept transactions where users have a history of at least 250 submitted posts or comments. The resulting network from scrapping contains 248 nodes (Reddit users) and 6222 transactions. They used 5722 transactions to train the model and then 500 for testing.

After obtaining the dataset, the authors first carry out a similar analysis as they do for Enron email dataset. They run the inference algorithm on the data and calculate BIC values for different numbers of groups K. The best K is 6 in this case, so they provide analysis when K=6. Entries in the estimated B matrix are much smaller than those in the B matrix from Enron dataset because this network contains more nodes and the average number of recipients per transaction is smaller. Unlike the B matrix from Enron dataset where the diagonal entries are significantly larger than non-diagonal entries, the B matrix for Reddit dataset has large values on both diagonal and non-diagonal entries. We can learn about sending and receiving behaviors from the B matrix. For example, group 1 does not respond to posts and only makes posts. One problem with the B matrix is that it fails to take into account different numbers of people in each group. If we just look at the B matrix, we have the impression that group 3 responds to many posts. However, there are only five people in group 3. After multiplying entries in the B matrix with the expected number of members in each group, group 3 is no longer as active as it seems to be in the B matrix. In the group-size-weighted B matrix, we learn that group 5 and 6 tend to interact with people within the same group. Also, group 4 tends to get more responses from people outside of the group. There is no information in the paper that explains how groups identified by TMMSB relates to the true subreddit groups of users. It also does not provide any practical interpretations for these groups.

Following that, the authors also examine predictive performance using the clustering and link prediction metrics we explained before. In order to see how well a method ranks the true recipients, the authors use the value of the rank at 100 % of recall. A small rank means that the model is able to identify all true recipients before many non-recipients are identified. Again, the authors use predicted probability $\text{P}(j \text{ receives} \mid i \text{ sends}) = p_{ij} = \pi_i B \pi_j^T$ to rank nodes. They use the rank of the last predicted recipient as performance measurement for that transaction and define the overall performance to be the average of individual performances of all messages. They then compare link prediction results produced by TMMSB, MMSB and hierarchical clustering and demonstrate that TMMSB has a significantly better performance when K = 6. Hierarchical clustering method generates class labels using the frequency matrix and assigns each node a fixed group. MMSB assigns mixed membership vectors to nodes but does not take into account the co-recipient and frequency information. Different from hierarchical clustering and MMSB, TMMSB allows for mixed memberships and incorporates frequencies and co-recipient information and thus has a superior performance.

2 Simulation Results

In this section, we simulate networks using the generative process outlined in the paper and recover B matrix using the variational Bayesian inference algorithm. Next, we compare the result from TMMSB to that of MMSB. Finally, we discuss our simulation results.

2.1 Generative Process

In the paper, the authors provide one detailed example about how they simulate the network and recover the B matrix. We decided to follow their parameter settings and reproduced the process. We set M = 65, N = 650, $\alpha = 0.25$, K = 4. True B matrix is defined as follows:

We then followed Algorithm 1 to simulate transactional data and obtained a Y matrix with 650 rows and 65 columns. We then transformed the raw transactional data into the 65×65 adjacency matrix by setting the threshold of counts as 1. In Figure 2, we plotted the adjacency matrix in which black cells indicate 1's and white cells indicate 0's.

0.01	0.2	0.01	0.01
0.01	0.3	0.2	0.1
0.1	0.01	0.01	0.3
0.1	0.01	0.01	0.3

Table 1: *True B matrix*

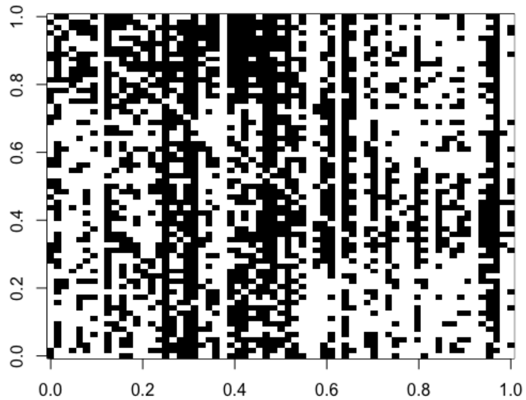

Figure 2: *Adjacency matrix for true network*

2.2 Inference

Because the authors did not specify how they initialised the B matrix, we tried different ways to initialise the B matrix. They also did not specify the convergence condition in the paper, so we check the change of log likelihood for convergence.

For the E-step, we calculated log likelihood after each iteration and found that given a B matrix, the log likelihood no longer changes after 100 iterations (see Figure 3). The reason why we used log likelihood instead of likelihood is that likelihood value is too small to evaluate. So we ran the E-step for 100 iterations.

For the M-step, we tried three ways of initializing B matrix: random B, true B and paper B. For the random B, we randomly sampled each entry from $Beta(2, 2)$ distribution. For the true B, we used the ground truth B matrix as the initial B matrix. For the paper B, we used the estimated B matrix listed in the paper (Table 2, right, in the paper) as the starting B matrix. To check for convergence, we calculated and plotted log likelihood. Below, we present our estimated B matrix and log likelihood graphs using different initialized B matrix.

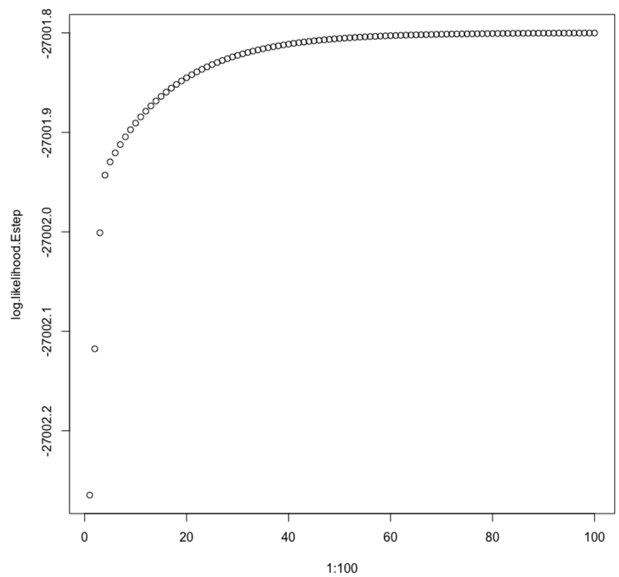

Figure 3: *Log likelihood plot for E-step*

For random B, after around 20 iterations, the log likelihood became stable and did not change anymore (see Figure 4). Therefore, we stopped the inference algorithm and produced the estimated B matrix (see Table 2). However, the estimated B matrix is very far away from the true B matrix.

0.0875	0.4647	0.3421	0.4512
0.2436	0.7435	0.6339	0.6591
0.0023	0.0181	0.0114	0.0160
0.0254	0.1777	0.1179	0.1599

Table 2: *Estimated B matrix with random B*

For true B, after around 30 iterations, the log likelihood seemed to become stable (see Figure 5), but after around 60 iterations, the log likelihood suddenly became larger (see Figure 6). It seemed to stablize after around 80 iterations. We recorded the estimated B matrix after 30 iterations (see Table 3) and after 80 iterations (see Table 4). Both of them were closer to the true B matrix than the estimated B matrix from the randomly initialised B. However, they did not look like the estimated B matrix presented in the paper.

For paper B, after around 50 iterations, the log likelihood seemed to become stable (see Figure 7), but after around 60 iterations (see Figure 8), the value of log likelihood started

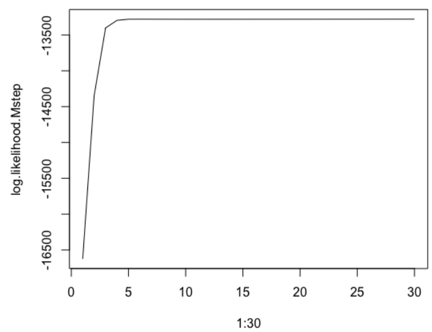

Figure 4: *Log likelihood plot for M-step with random B*

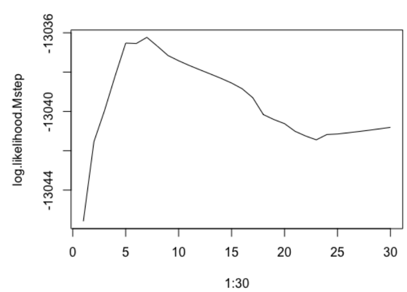

Figure 5: *Log likelihood plot for M-step with true B after 30 iterations*

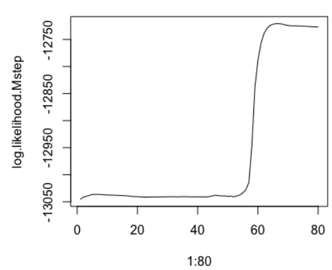

Figure 6: *Log likelihood plot for M-step with true B after 80 iterations*

0.0022	0.0835	0.0179	0.0198
0.0064	0.2057	0.1373	0.1684
0.1391	0.0186	0.0149	0.3843
0.1210	0.0197	0.0140	0.4022

Table 3: *Estimated B matrix with true B after 30 iterations*

0.0190	0.1514	0.0339	0.0037
0.0295	0.2441	0.1635	0.1075
0.0837	0.0568	0.0389	0.1890
0.1021	0.0094	0.0313	0.2686

Table 4: *Estimated B matrix with true B after 80 iterations*

to increase. We recorded the estimated B matrix after 50 iterations (see Table 5) and after 70 iterations (see Table 6). Both of estimated B matrices were different than the true B matrix and the estimated B matrix presented in the paper.

2.0448e-02	0.2851	0.0071	0.0014
1.9514e-05	0.5165	0.2184	0.2129
8.8604e-02	0.0118	0.0269	0.2395
1.0289e-01	0.0123	0.0339	0.2715

Table 5: *Estimated B matrix with paper B after 50 iterations*

1.9912e-02	0.2945	0.0078	0.0008
4.7713e-05	0.5630	0.2207	0.1998
8.1792e-02	0.0081	0.0357	0.2038
9.2095e-02	0.0088	0.0475	0.2342

Table 6: *Estimated B matrix with paper B after 70 iterations*

2.3 Comparison with MMSB

In addition, we investigated whether the inclusion of transactional information in the TMMSB could improve model performance. As a comparison, we fitted standard MMSB model with $K = 4$ groups on the simulated network with 100000 iterations, using LDA package. The estimated B matrix is shown in Table 7. Entries in this B matrix are significantly larger than those in the true B matrix.

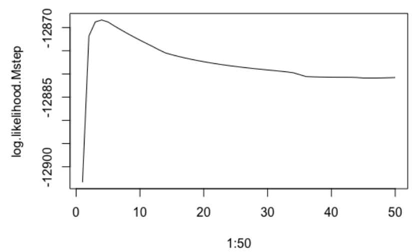

Figure 7: *Log likelihood plot for M-step with paper B after 50 iterations*

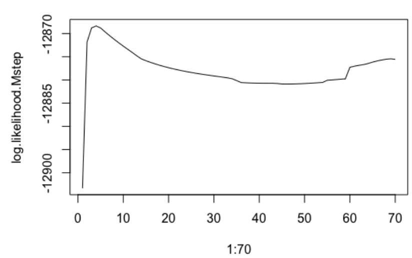

Figure 8: *Log likelihood plot for M-step with paper B after 70 iterations*

0.7	0.3175	0.1778	0.2837
0.9925	0.9809	0.8319	0.9386
0.6120	0.5932	0.0274	0.4708
0.0206	0.009	0	0.0029

Table 7: *Estimated B matrix with MMSB after 100,000 iterations*

2.4 Discussion

Among three methods of initializing B matrix, we found that setting it as the ground truth B matrix gave us the most accurate estimation, although the log likelihood is not guaranteed to converge. Initializing B matrix randomly ensures that log likelihood becomes stable quickly, but it does not recover the true B matrix. Using the estimated B matrix given in the paper as the starting point performs poorly in terms of log likelihood and produces small values for some entries in the estimated B matrix. After experimenting with the algorithm, we were not able to recover the true matrix with the algorithm presented in the paper. It is possible that authors did not include enough information in the paper for us to reproduce the result. It is also possible that the algorithm is not reproducible or has other problems.

References

[1] Shafiei, M. and Chipman, H. *Mixed membership stochastic block-models for transactional networks.* Proc. 10th Int. Conf. Data Mining, pp. 1019-1024, 2010.

[2] Airoldi, E., Blei, D., Fienberg, S., and Xing, E. *Mixed membership stochastic blockmodels.* Journal of Machine Learning Research, 9:1981-2014, 2008.

[3] Amig, E., Gonzalo, J., Artiles, J., and Artiles, F. *A comparison of extrinsic clustering evaluation metrics based on formal constraints.* Information Retrieval, 12(4):461-486, 2009.

[4] Kahanda, I. and Neville J. *Using transactional information to predict link strength in online social networks.* Proceedings of the 4th International AAAI Conference on Weblogs and Social Media.

[5] Kurihara, K., Kameya, Y., and Sato, T. *A frequency-based stochastic blockmodel.* In Workshop on Information Based Induction Sciences, 2006.