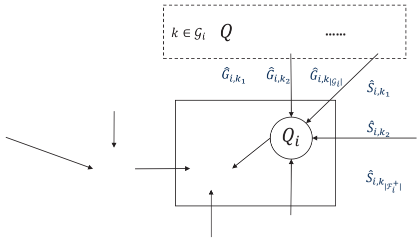


$k \in \mathcal{G}_i$
$Q$
......
$\hat{G}_{i,k_1}$
$\hat{G}_{i,k_2}$
$\hat{G}_{i,k_{|\mathcal{G}_i|}}$
$\hat{S}_{i,k_1}$
$\hat{S}_{i,k_2}$
$\hat{S}_{i,k_{|\mathcal{F}_i^+|}}$
$Q_i$

similar users and similar votings. We give detailed explanation as follows. The first term of Eq. (17) measures the mean squared error between prediction and ground truth, where $I'_{i,j}$ is the training weights defined as

$$I'_{i,j} = \begin{cases} 1, & if\ u_i\ participates\ v_j \\ I_m, & otherwise \end{cases} . \tag{18}$$

The reason we do not directly use $I_{u_i,v_j}$ defined in Eq. (1) as the training weights is because we found a small and positive $I_m$ makes the training process more robust and can greatly improve the results. $R_{i,j}$ is the actual rating of user $u_i$ on voting $v_j$, and $\boldsymbol{Q_i P_j^\top}$ is the predicted value of $R_{i,j}$. Without loss of generality, in JTS-MF model,

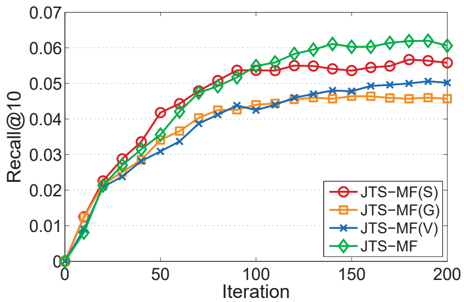



**Fig. 6: Convergence of JTS-MF models with respect to *Recall@10*.**

## 7.2 Parameter Settings

We use GibbsLDA++[8], an open-source implementation of LDA using Gibbs sampling, to calculate topic information of words and documents in JTS-MF and Topic-MF models. We set the number of topics to 50 and leave all other parameters in LDA as default values. For word embeddings in JTS-MF and Semantic-MF models, we use the same settings as follows: length of embedding dimension as 50, window size as 5, and number of negative samples as 3.

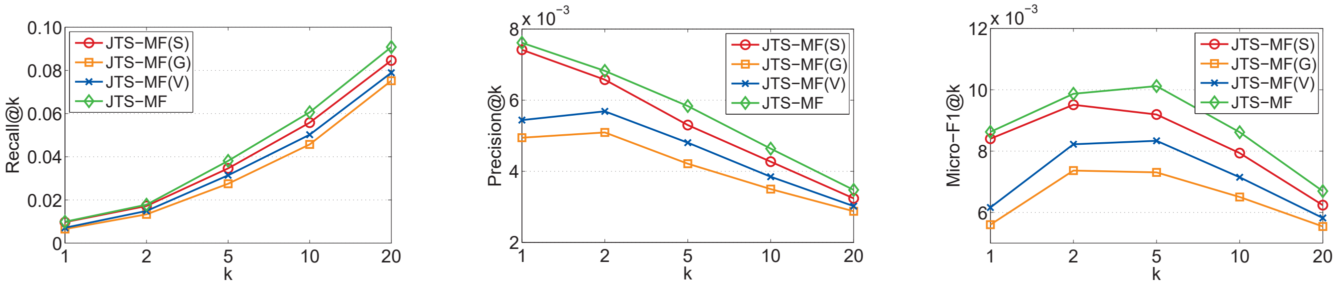
For all MF-based methods, we set the learning rate $\delta = 0.001$ and regularization weight $\lambda = 0.5$ by 10-fold cross validation. Typically, we set $I_m = 0.01$ in Eq. (18). Taking into consideration the balance of experimental results and time complexity, we run 200 iterations for each of the experiment cases. To conduct the recommendation task, we randomly select 20% of users' voting records in the dataset as test set and use the remaining data as the trainning examples for our JTS-MF model as well as all baselines. The choice of remaining hyper-parameters (trade-off parameters $\alpha$, $\beta$, $\gamma$, and dimension of latent features $dim$) is discussed in Section 7.4.

To quantitatively analyze the performance of voting recommendation, in our experiment, we use *top-k recall* (*Recall@k*), *top-k precision* (*Precision@k*), and *top-k micro-F1* (*Micro-F1@k*) as the evaluation metrics.

## 7.3 Experiment Results

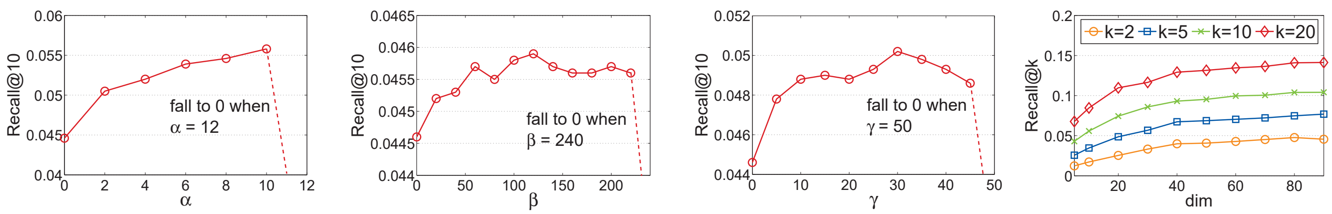
*7.3.1 Study of convergence.* To study the convergence of JTS-MF model, we run the learning algorithm up to 200 iterations for JTS-MF(S) with $\alpha = 10$, JTS-MF(G) with $\beta = 140$, JTS-MF(V) with $\gamma = 30$, JTS-MF with $\alpha = 10$, $\beta = 140$, $\gamma = 30$ ($dim = 10$ for $\boldsymbol{Q_i}$ and $\boldsymbol{P_j}$ in all models), then calculate *Recall@10* for every 10 iterations. The result of convergence of JTS-MF models is plotted in Fig. 6. From Fig. 6 we can see that, the recall of JTS-MF models rises rapidly before 100 iterations, and starts to oscillate slightly after around 150 iterations. The same changing pattern is observed for all four JTS-MF variants. Therefore, we set the number of learning iterations as 200 to achieve a balance between running time and performance of models.

*7.3.2 Study of JTS-MF.*




(a) (b) (c)

**Fig. 7: (a) *Recall@k*, (b) *Precision@k*, and (c) *Micro-F1@k* of JTS-MF models.**




(a) (b) (c) (d)

**Fig. 8: Parameter sensitivity with respect to (a) $\alpha$, (b) $\beta$, (c) $\gamma$, and (d) $dim$.**

**Table 2: Result of *Recall@k*, *Precision@k*, and *Micro-F1@k* for JTS-MF model and baselines.**

| Model | Metric | k=1 | 2 | 5 | 10 | 20 | 50 | 100 | 500 |
|---|---|---|---|---|---|---|---|---|---|
| JTS-MF(S) | *Recall* | 0.0097 | 0.0172 | 0.0346 | 0.0558 | 0.0846 | **0.1529** | **0.2229** | **0.4392** |
| | *Precision* | 0.007416 | 0.006575 | 0.005300 | 0.004271 | 0.003238 | **0.002341** | **0.001707** | **0.000672** |
| | *Micro-F1* | 0.008401 | 0.009511 | 0.009192 | 0.007935 | 0.006238 | **0.004612** | **0.003387** | **0.001343** |
| JTS-MF(G) | *Recall* | 0.0065 | 0.0133 | 0.0275 | 0.0457 | 0.0752 | 0.1360 | 0.2051 | 0.4216 |
| | *Precision* | 0.004944 | 0.005092 | 0.004212 | 0.003500 | 0.002877 | 0.002082 | 0.001570 | 0.000645 |
| | *Micro-F1* | 0.005601 | 0.007365 | 0.007306 | 0.006503 | 0.005542 | 0.004102 | 0.003116 | 0.001289 |
| JTS-MF(V) | *Recall* | 0.0071 | 0.0149 | 0.0314 | 0.0502 | 0.0789 | 0.1387 | 0.2049 | 0.4176 |
| | *Precision* | 0.005439 | 0.005685 | 0.004805 | 0.003846 | 0.003021 | 0.002124 | 0.001568 | 0.000639 |
| | *Micro-F1* | 0.006161 | 0.008223 | 0.008335 | 0.007145 | 0.005819 | 0.004184 | 0.003112 | 0.001277 |
| JTS-MF | *Recall* | **0.0099** | **0.0178** | **0.0381** | **0.0606** | **0.0908** | 0.1520 | 0.2187 | 0.4297 |
| | *Precision* | **0.007614** | **0.006823** | **0.005834** | **0.004637** | **0.003475** | 0.002327 | 0.001674 | 0.000658 |
| | *Micro-F1* | **0.008625** | **0.009868** | **0.010118** | **0.008615** | **0.006695** | 0.004585 | 0.003322 | 0.001314 |
| MostPop | *Recall* | 0.0042 | 0.0085 | 0.0191 | 0.0313 | 0.0517 | 0.0974 | 0.1455 | 0.3086 |
| | *Precision* | 0.003221 | 0.003261 | 0.002921 | 0.002403 | 0.001972 | 0.001482 | 0.001119 | 0.000469 |
| | *Micro-F1* | 0.003637 | 0.004721 | 0.005062 | 0.004468 | 0.003804 | 0.002925 | 0.002218 | 0.000937 |
| Basic-MF | *Recall* | 0.0063 | 0.0129 | 0.0274 | 0.0446 | 0.0727 | 0.1368 | 0.2050 | 0.4198 |
| | *Precision* | 0.004845 | 0.004944 | 0.004192 | 0.003411 | 0.002783 | 0.002094 | 0.001569 | 0.000643 |
| | *Micro-F1* | 0.005489 | 0.007151 | 0.007271 | 0.006337 | 0.005361 | 0.004125 | 0.003114 | 0.001283 |
| Topic-MF | *Recall* | 0.0076 | 0.0147 | 0.0311 | 0.0495 | 0.0781 | 0.1395 | 0.2076 | 0.4210 |
| | *Precision* | 0.005834 | 0.005636 | 0.004766 | 0.003787 | 0.002991 | 0.002136 | 0.001589 | 0.000644 |
| | *Micro-F1* | 0.006609 | 0.008152 | 0.008266 | 0.007035 | 0.005761 | 0.004207 | 0.003154 | 0.001287 |
| Semantic-MF | *Recall* | 0.0093 | 0.0169 | 0.0333 | 0.0545 | 0.0860 | 0.1471 | 0.2142 | 0.4293 |
| | *Precision* | 0.007120 | 0.006476 | 0.005102 | 0.004173 | 0.003293 | 0.002252 | 0.001639 | 0.000657 |
| | *Micro-F1* | 0.008065 | 0.009368 | 0.008849 | 0.007752 | 0.006342 | 0.004437 | 0.003254 | 0.001313 |

parameter. Then we report *Recall@10* in Fig. 8a, 8b, and 8c, respectively.

As shown in Fig. 8a, the *Recall@10* increases constantly as $\alpha$ gets larger and reaches a maximum of 0.0558 when $\alpha = 10$. This suggests that the usage of users' social-level similarity do help to improve the recommendation performance. However, when $\alpha$ is too large ($\alpha =$ 12), the learning algorithm of JTS-MF is misled to wrong direction when updating latent features of users and votings, resulting in performance deterioration. The similar phenomenon are observed in Fig. 8b and Fig. 8c, too. According to the results, when the other two trade-off parameters are set to 0, *Recall@10* reaches the maximum when $\alpha = 10$, $\beta = 140$, and $\gamma = 30$, respectively.

Therefore, in previous experiments we adopt these optimal settings for JTS-MF(S), JTS-MF(G), and JTS-MF(V), respectively, and use their combination as the parameter settings in JTS-MF.

*7.4.2 Dimension of latent features.* We fix $\alpha = 10$, $\beta = 0$, $\gamma = 0$ and tune the dimension of latent features of users and votings from 5 to 90. The result is shown in Fig. 8d. From the figure, we can see clearly that the recall is increasing when $dim$ gets larger, this is because latent features with larger number of dimensions have more capacity to characterize users and votings. But a larger $dim$ leads to more running time in experiments. Moreover, we notice that the improvement of performance stagnates after $dim$ reaches 80. On balance, we set $dim = 10$ in our experiment scenarios to ensure the experiments can complete within rational time duration.

## 8 CONCLUSIONS

In this paper, we study the problem of recommending online votings to users in social networks. We first formalize the voting recommendation problem and justify the motivation of leveraging social structure and voting content information. To overcome the limitations of topic models and semantic models when learning representation of voting content, we propose Topic-Enhanced Word Embedding method to jointly consider topics and semantics of words and documents. We then propose our Joint-Topic-Semantic-aware social Matrix Factorization model to learn latent features of users and votings based on the social network structure and TEWE representation. We conduct extensive experiments to evaluate JTS-MF with Weibo voting dataset. The experimental results prove the competitiveness of JTS-MF against other state-of-the-art baselines and demonstrate the efficacy of TEWE representation.

## ACKNOWLEDGMENTS

This work was partially sponsored by the National Basic Research 973 Program of China under Grant 2015CB352403, the NSFC Key Grant (No. 61332004), PolyU Project of Strategic Importance 1-ZE26, and HK-PolyU Grant 1-ZVHZ.

## REFERENCES


[1] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. In *Journal of Machine Learning Research*. 993–1022.

[2] Jesús Bobadilla, Fernando Ortega, Antonio Hernando, and Abraham Gutiérrez. 2013. Recommender systems survey. *Knowledge-Based Systems* 46 (2013).

[3] Marco Bressan, Stefano Leucci, Alessandro Panconesi, Prabhakar Raghavan, and Erisa Terolli. 2016. The Limits of Popularity-Based Recommendations, and the Role of Social Ties. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 745–754.

[4] Danqi Chen and Christopher D Manning. 2014. A Fast and Accurate Dependency Parser using Neural Networks.. In *EMNLP*. 740–750.

[5] Qiming Diao, Minghui Qiu, Chao-Yuan Wu, Alexander J Smola, Jing Jiang, and Chong Wang. 2014. Jointly modeling aspects, ratings and sentiments for movie recommendation (jmars). In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 193–202.

[6] Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu. 2015. Content-aware point of interest recommendation on location-based social networks.. In *AAAI*. 1721–1727.

[7] Thomas L. Griffiths and Mark Steyvers. 2004. Finding scientific topics. In *Proceedings of National Academy of Sciences (PNAS)*. 5228–5235.

[8] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 855–864.

[9] Liang Hu, Jian Cao, Guandong Xu, Longbing Cao, Zhiping Gu, and Can Zhu. 2013. Personalized recommendation via cross-domain triadic factorization. In *Proceedings of the 22nd international conference on World Wide Web*. ACM, 595–606.

[10] Hui-Ju Hung, Hong-Han Shuai, De-Nian Yang, Liang-Hao Huang, Wang-Chien Lee, Jian Pei, and Ming-Syan Chen. 2016. When Social Influence Meets Item Inference. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 915–924.

[11] Janet Ilieva, Steve Baron, and Nigel M Healey. 2002. Online surveys in marketing research: Pros and cons. *International Journal of Market Research* 44, 3 (2002), 361.

[12] Meng Jiang, Peng Cui, Xumin Chen, Fei Wang, Wenwu Zhu, and Shiqiang Yang. 2015. Social recommendation with cross-domain transferable knowledge. *IEEE Transactions on Knowledge and Data Engineering* 27, 11 (2015), 3084–3097.

[13] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009).

[14] Ken Lang. 1995. Newsweeder: Learning to filter netnews. In *Proceedings of the 12th international conference on machine learning*. 331–339.

[15] Quoc V Le and Tomas Mikolov. 2014. Distributed Representations of Sentences and Documents.. In *ICML*, Vol. 14. 1188–1196.

[16] Wu-Jun Li, Dit-Yan Yeung, and Zhihua Zhang. 2011. Generalized latent factor models for social network analysis. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI), Barcelona, Spain*. 1705.

[17] Bing Liu and Lei Zhang. 2012. A survey of opinion mining and sentiment analysis. In *Mining text data*. Springer, 415–463.

[18] Yang Liu, Zhiyuan Liu, Tat-Seng Chua, and Sun Maosong. 2015. Topical word embeddings. In *Proceedings of the Twenty-Ninth Conference on Artificial Intelligence (AAAI)*. AAAI, 2418–2424.

[19] Hao Ma, Haixuan Yang, Michael R Lyu, and Irwin King. 2008. Sorec: social recommendation using probabilistic matrix factorization. In *Proceedings of the 17th ACM conference on Information and knowledge management*. ACM, 931–940.

[20] Qiaozhu Mei, Xu Ling, Matthew Wondra, Hang Su, and ChengXiang Zhai. 2007. Topic sentiment mixture: modeling facets and opinions in weblogs. In *Proceedings of the 16th international conference on World Wide Web*. ACM, 171–180.

[21] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems (NIPS)*. 3111–3119.

[22] Steffen Rendle and Christoph Freudenthaler. 2014. Improving pairwise learning for item recommendation from implicit feedback. In *Proceedings of the 7th ACM international conference on Web search and data mining*. ACM, 273–282.

[23] Sam T Roweis and Lawrence K Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *science* 290, 5500 (2000), 2323–2326.

[24] Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey Hinton. 2007. Restricted Boltzmann machines for collaborative filtering. In *Proceedings of the 24th international conference on Machine learning*. ACM, 791–798.

[25] Yue Shi, Alexandros Karatzoglou, Linas Baltrunas, Martha Larson, Nuria Oliver, and Alan Hanjalic. 2012. CLiMF: learning to maximize reciprocal rank with collaborative less-is-more filtering. In *Proceedings of the sixth ACM conference on Recommender systems*. ACM, 139–146.

[26] Shaojie Tang and Jing Yuan. 2016. Optimizing Ad Allocation in Social Advertising. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. ACM, 1383–1392.

[27] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. 2015. Collaborative deep learning for recommender systems. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1235–1244.

[28] Xin Wang, Roger Donaldson, Christopher Nell, Peter Gorniak, Martin Ester, and Jiajun Bu. 2016. Recommending groups to users using user-group engagement and time-dependent matrix factorization. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. AAAI Press, 1331–1337.

[29] Xin Wang, Wei Lu, Martin Ester, Can Wang, and Chun Chen. 2016. Social Recommendation with Strong and Weak Ties. In *Proceedings of the 25th ACM CIKM*. ACM, 5–14.

[30] Zhonghang Xia, Yulin Dong, and Guangming Xing. 2006. Support vector machines for collaborative filtering. In *Proceedings of the 44th annual Southeast regional conference*. ACM, 169–174.

[31] Xiwang Yang, Chao Liang, Miao Zhao, Hongwei Wang, Hao Ding, Yong Liu, Yang Li, and Junlin Zhang. 2017. Collaborative filtering-based recommendation of online social voting. *IEEE Transactions on Computational Social Systems* 4, 1 (2017), 1–13.

[32] Qin Zhang, Jia Wu, Peng Zhang, Guodong Long, Ivor W Tsang, and Chengqi Zhang. 2016. Inferring Latent Network from Cascade Data for Dynamic Social Recommendation. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*. IEEE, 669–678.

[33] Wayne Xin Zhao, Jing Jiang, Hongfei Yan, and Xiaoming Li. 2010. Jointly modeling aspects and opinions with a MaxEnt-LDA hybrid. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 56–65.

[34] Hengshu Zhu, Enhong Chen, Kuifei Yu, Huanhuan Cao, Hui Xiong, and Jilei Tian. 2012. Mining personal context-aware preferences for mobile users. In *2012 IEEE 12th International Conference on Data Mining*. IEEE, 1212–1217.