the MF model. For example in Fig. 3, comparing with the MF model, the DNN based model lowers the error by 2.47% for the basic model, 2.53% when sentiment scores are added as a rating matrix, 1.75% when sentiment scores are added as a rating matrix with other features, and 3.5% when we add all the trailer feedback data as movie features.

We compare our best-performing DNN model (using all the trailer feedback data as features) with some of the baseline algorithms, including the basic MF model, SVD, SVD++, NMF [15], NCF [8], RBM [16] and SAR [17]. For all of the baselines, we take their default parameters. Fig. 5 shows the comparison based on the RMSE scores.

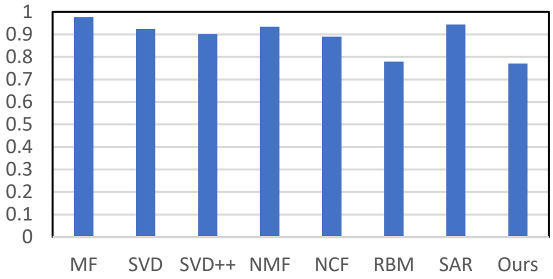

Fig. 5. Comparing with baseline algorithms using RMSE

From the figure, we see that our model has the smallest RMSE score compared to the baselines. Out of all the baseline methods, RBM has the lowest error rate and our model lowers that error by 1.18%. Fig. 6 shows the comparison of our model with these baseline algorithms on F1@10.

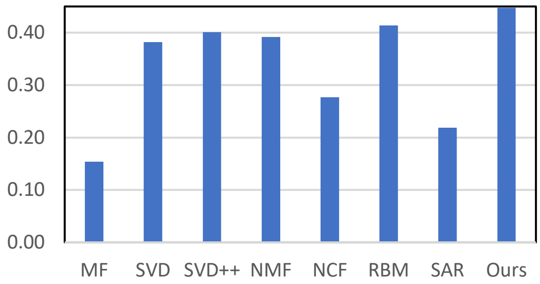

Fig. 6. Comparing with baseline algorithms using F1@10

The figure shows that our model has the most accurate result compared to the other models in terms of the F1-scores. Out of the matrix factorization models, SVD++ generates the most accurate result while our model improves it by 11.5%. Out of the neural network models, RBM generates the most accurate result, while our model improves it by 8.1%.

In terms of the running time, for our dataset, MF-based model takes longer time than the DNN-based model. When adding the trailer feedback data, we record longer running time (~35% longer) from both models compared to the case when the original model is used without the side information. This is expected considering the extra processing required.

V. CONCLUSION AND FUTURE WORK

This work evaluates the effectiveness of adding movie trailer feedbacks as the side information to the movie rating data for movie recommendation. To integrate the trailer data, we have used three approaches: integrating all of them as movie features; treating sentiment scores as a rating matrix to integrate with the movie rating matrix and others as the movie features; only integrating the sentiment rating matrix with the movie rating matrix. Overall, the evaluation results show that if we include movie trailer data, it reduces the prediction error and increases the recommendation accuracy. As for the way of integration, if all the trailer feedback data is integrated as the movie features, our recommender system provides the most accurate result. We also find that deep neural network model performs better than the matrix factorization model.

In future, we want to extend our system to add temporal signals as the side information. As users' criteria to find a movie and user preferences on a movie may change over time, if we can add the temporal signals in our model, we might be able to recommend movies based on users' changing interests. We also want to try different DNN models to implement our recommender system. For example, instead of MLP, we may try the CNN model, or the RNN model if we consider the sequence information (time of ratings).

REFERENCES

[1] F. M. Harper and J. A. Konstan, The movielens datasets: History and context, Acm transactions on Interactive Intelligent Systems, vol. 5, no. 4, pp. 1-19, 2015.

[2] J. Bennett and S. Lanning, The netflix prize, In the Proceedings of KDD Cup and Workshop, pp. 35, 2007.

[3] IMDb, [Online]. Available: https://datasets.imdbws.com/. [Accessed 1 12 2019].

[4] P. Lops, M. de Gemmis and G. Semeraro, Recommender systems handbook, Springer, pp. 1-186, 2011.

[5] F. Li, G. Xu and L. Cao,Two-level matrix factorization for recommender systems, Neural Computing and Applications, vol. 27, no. 8, pp. 2267-2278, 2016.

[6] Z. Sun, L. Han, W. Huang, X. Wang, X. Zeng, M. Wang and H. Yan, Recommender systems based on social networks, Journal of Systems and Software, vol. 99, pp. 109-119, 2015.

[7] J. Davidson, B. Liebald, J. Liu, P. Nandy, T. Van Vleet, U. Gargi, S. Gupta, Y. He, M. Lambert, B. Livingston and D. Sampath,The YouTube video recommendation system, In the Proceedings of the fourth ACM conference on Recommender systems, pp. 293-296, 2010.

[8] X. He, L. Liao, H. Zhang, L. Nie, X. Hu and T. S. Chua, Neural collaborative filtering, In the Proceedings of the 26th international conference on world wide web, pp. 173-182, 2017.

[9] H. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir and R. Anil, Wide & deep learning for recommender systems, In the Proceedings of the 1st workshop on deep learning for recommender systems, pp. 7-10, 2016.

[10] H. Guo, R. Tang, Y. Ye, Z. Li and X. He, DeepFM: a factorization-machine based neural network for CTR prediction, arXiv preprint arXiv:1703.04247, 2017.

[11] A. Farseev, L. Nie, M. Akbari and T. S. Chua, Harvesting multiple sources for user profile learning: a big data study, In the Proceedings of the 5th ACM on International Conference on Multimedia Retrieval, pp. 235-242, 2015.

[12] M. Yan, J. Sang and C. Xu, Unified YouTube video recommendation via cross-network collaboration, In the Proceedings of the 5th ACM on International Conference on Multimedia Retrieval, pp. 19-26, 2015.

[13] J. Zhu, J. Zhang, L. He, Q. Wu, B. Zhou, C. Zhang and P. S. Yu, Broad Learning based Multi-Source Collaborative Recommendation, In the Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp. 1409-1418, 2017.

[14] C. J. Hutto and E. Gilbert, Vader: A parsimonious rule-based model for sentiment analysis of social media text., In Eighth international AAAI conference on weblogs and social media, 2014.

[15] H. Lee, J. Yoo and S. Choi, Semi-supervised nonnegative matrix factorization, IEEE Signal Processing Letters, 17(1), pp. 4-7, 2009.

[16] R. Salakhutdinov and A. a. H. G. Mnih, Restricted Boltzmann machines for collaborative filtering, In the Proceedings of the 24th international conference on Machine learning, pp. 791-798, 2007.

[17] B. Sarwar, G. Karypis and J. a. R. J. Konstan, Item-based collaborative filtering recommendation algorithms, In the Proceedings of the 10th international conference on World Wide Web, pp. 285-295, 2001.