EasyChair Preprint

№ 6743

Investigating the Effects of Pre-Trained BERT to Improve Sparse Data Recommender Systems

Nguyen Huy Xuan, Le Minh Nguyen and Long H. Trieu

October 3, 2021

Investigating the Effects of Pre-trained BERT to Improve Sparse Data Recommender Systems

No Author Given

No Institute Given

Abstract.

Recommender systems play an important role with many applications in natural language processing such as in e-commerce services. Matrix factorization (MF) is a powerful method in recommender systems, but a main issue is the sparse data problem. In order to overcome the problem, some previous models use neural networks to represent additional information such as product item reviews to enhance MF-based methods, and obtain improvement in recommender systems. However, these models use conventional pre-trained word embeddings, which raise a question whether recent powerful models such as pre-trained BERT can improve these MF-based methods enhanced by item reviews. In this work, we investigate the effect of utilizing pre-trained BERT model to improve some previous models, especially focusing on several specific sparse data settings. Experimental results and intensive analyses on the MovieLens dataset show some promising findings for our model, which open directions to improve this on-going model to solve the problem of sparse data in MF-based recommender systems.

1 Introduction

Recommendation systems play an important role in natural language processing applications such as e-commerce, e-learning, e-business services which cover various domains such as recommending books, movies, documents, etc [1, 3]. One of the most effective methods for recommendation systems is called *collaborative filtering (CF)* [5, 12]. Given users and product items and the relationships among users and items such as ratings (for instance a score from 1 to 5 a user gives to a product), CF identify new relationships among users and items based on existing relationships. In CF methods, *matrix factorization (MF)* [7] is a powerful method and commonly used recently, which tries to explain the ratings by characterizing items and users by vectors of latent *factors* inferred from the ratings patterns (for instance, drama versus comedy, amount of action, etc in discovered *factors* of movie domain). One of the main issues of CF methods is the sparseness data problem when majority of items may lack the feedback(or ratings) from users [4, 8].

In order to overcome the sparseness issue in CF methods, external information can be utilized to enhance new rating prediction such as item reviews

(textual comments that a user gives to product items) [6, 7]. Item reviews can be represented by convolutional neural networks (CNNs) then combined with a probabilistic matrix factorization model [6] (ConvMF). Instead of using CNNs, a recent model called AMF [10] improves the ConvMF based on an attention mechanism with genre information of product items. However, these models are based on conventional pre-trained word embeddings such as Glove [11] while recent proposed pre-trained models such as BERT [2] are still yet investigated.

In this work, we propose a model for recommender system which combines the probabilistic matrix factorization enhanced by contextual information of item reviews represented by utilizing pre-trained BERT models. Our goal is to investigate the effect of pre-trained BERT models to improve the previous document-enhanced matrix factorization based model [10]. In this model, we utilize pre-trained BERT models [2] for item review representations instead of the pre-trained word embeddings such as Glove [11] used in the AMF model [10]. We evaluate our model on the widely used MovieLens-1m dataset and compare with the baseline AMF model [10] as well as with some other previous models including the ConvMF[6]. In addition, we conduct intensive analyses to investigate our models on different aspects of sparse data, a challenge which is still remaining for recommender systems. Experimental results show that our model obtains better performance than the ConvMF, but still lower than the AMF model. However, our findings from the analyses are that our model improves the baseline AMF model with the data setting where review text lengths are in a specific range (less than 200 words), which may open a direction for our model to deal with sparse data issues.

2 Our model

The overall architecture of our model is presented in Figure 1. The model consists of two components: the probabilistic matrix factorization (PMF) and the item review representations based on BERT.

2.1 Probabilistic Matrix Factorization (PMF)

Matrix factorization is one of the most popular methods in collaborative filtering-based (CF) for recommender systems [7]. PMF models can learn low-rank representations (latent factors) of users and items from the user-item matrix, which are then used to predict new ratings between users and items. Given N is the set of users, M is the set of items, and R is a rating matrix of users for items ($R \in \mathbb{R}^{N \times M}$). PMF discovers the k-dimensional models, which are the latent models of user u_i ($u_i \in \mathbb{R}^k$) and item v_j ($v_j \in \mathbb{R}^k$). The rating r_{ij} of user i on item j can be approximated by equation: $r_{ij} \approx \hat{r}_{ij} = {u_i}^T v_j$.

2.2 Representations of item reviews

In this section, we present the representations of item reviews based on pre-trained BERT models in our BMF model. Given reviews of each product item,

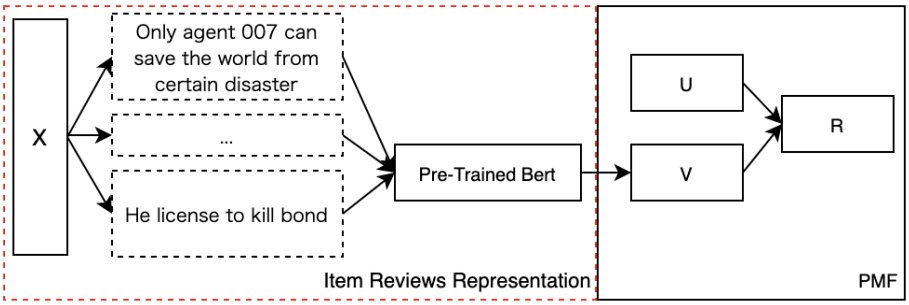

Fig. 1. The overall architecture of our model. The probabilistic matrix factorization (PMF) is in the right, and the item review representations are in the left. An user u_i of the user set U gives a rating to a product item v_j of the product item set V, which results in the ratings matrix R. A product item in V may have reviews X which are comments from users about this product. We represent these reviews X by using pre-trained BERT models to enhance the prediction of new ratings for items in V.

the review texts are passed through a pre-trained BERT model to form a vector for the item reviews. The review representations are then fed into a pooling layer and an output layer to get the final prediction. In the previous work AMF model [10], pre-trained word embeddings such as Glove [11] are used to represent item reviews. Instead of that, in this BMF model, we represent item reviews based on a pre-trained BERT model [2]. In recent years, pre-trained BERT models have shown to be effective when they are integrated into neural architectures in many NLP applications. In this work, we aim at investigating whether we can improve the AMF baseline model by using the pre-trained BERT model.

3 Experiments

3.1 Data

We evaluate our models on the MovieLens dataset[1] (MovieLen-1M), which is widely used in this task. For item review information, we extracted from the IMDB.[2] For genre information, we extracted from item files (*_movies.dat). For evaluation, we randomly divided each dataset into three sets: training (80%), validation (10%), and test sets (10%), which we followed the previous work [10]. The training set contains at least one rating on each user and each item so that all users and items are included in PMF.

3.2 Baseline

We compared our proposed BMF with previous models:

[1] https://grouplens.org/datasets/movielens/
[2] http://www.imdb.com/

- **PMF** [13]: Probabilistic Matrix Factorization uses only user ratings for CF. This is a standard rating prediction model.
- **CTR** [14]: Collaborative Topic Regression combines collaborative filtering (PMF) and topic modeling (LDA) to use both ratings and documents.
- **CDL** [15]: Collaborative Deep Learning improves rating prediction by analyzing documents.
- **ConvMF** [6]: Convolutional Matrix Factorization uses convolutional neural networks to represent item reviews to enhance rating prediction accuracy.
- **AMF** [10]: This model uses an Attention mechanism into Matrix Factorization. It employed the item genre information in attention neural network to find out attended features from item reviews. This is the main baseline of our model.

For our BMF model, we used different pre-trained BERT models in the BMF model, which result in the three different variants of our BMF model.

- **BMF(bert-base-uncased)**: we used the pre-trained bert-base-uncased model[3] to represent item reviews and combine with a PMF framework.
- **BMF(robeta-bert)**: this is the same as the BMF(bert-base-uncased) model but we use another pre-trained BERT model, i.e. the robeta-bert [9].
- **BMF(albert-base-v1)**: this is the same as the BMF(bert-base-uncased) but we use another pre-trained BERT model, i.e. the albert-base-v1.[4]

3.3 Settings

We implemented our model on Pytorch. We set the latent dimensions (U and V) as 50 according to the previous work in [15] and initialized U, V randomly from 0 to 1. The models are evaluated based on the widely used root mean squared error (RMSE), which we followed the previous work [6, 10].

3.4 Results

Table 1 presents rating prediction error of our BMF model and the baselines. The results show that our BMF model obtains better performance than most of the baselines, i.e., ConvMF, CDL, CTR, PMF, in which our model improves 0.6% in comparison with the powerful ConvMF model. However, our BMF model performance is still lower than the AMF result. We present further analyses to compare our BMF model and the AMF model in Section 3.5.

3.5 Analyses and discussions

We conduct analyses on the different aspects to investigate our model on different settings related to sparse data.

[3] https://huggingface.co/bert-base-uncased
[4] https://huggingface.co/albert-base-v1

Table 1. Comparison of our BMF with other models on the ML-1m test set (RMSE score: lower is better; the best score is in bold; the score that is better than the ConvMF baseline is in underline)

Model	RMSE
ConvMF [6]	0.8578
PMF [13]	0.8961
CTR [14]	0.8968
CDL [15]	0.8876
AMF [10]	**0.8350**
BMF(bert-base-uncased)	0.8516
BMF(robeta-bert)	0.8516
BMF(albert-base-v1)	0.8515

The effect of genre information We added the *item genre information* as additional input for the pre-train BERT model. We compare the BMF model with and without using the *item genre information*. The results in Table 2 show that using *item genre information* does not improve the performance. This result indicates that item genre information may not be helpful in combination with the pre-trained BERT model in our proposed BMF model. It may be because the length of item genre texts is quite short (with several words). In future work, we need to conduct more experiments on other datasets with longer text sequences of item genres to further investigate the contribution of genre information in combination with the powerful pre-trained BERT model in building recommender systems.

Table 2. The effect of *item genre information (IGI)* (RMSE score: lower is better)

Model	With IGI	Without IGI
BMF(bert-base-uncased)	0.8520	0.8516
BMF(robeta-bert)	0.8515	0.8516
BMF(albert-base-v1)	0.8525	0.8515

The effect of sparse data We investigate the effect of sparse data by setting the training data with different ratios, in which we used only 20%, 40%, 60%, and 80% of training data to train the model. The results presented in Table 3 show that our BMF model outperforms the baseline ConvMF [6] in all of the data ratios. Meanwhile, the AMF model [10] achieves the best performance. The results indicate that our BMF model, which uses pre-trained BERT, is not better than the baseline AMF [6] using pre-trained word embeddings in the setting of sparse data. Further analyses and investigations in experiments as well as modifications in the BMF model architecture are needed to improve our BMF model in future work, especially in this sparse data setting which is still a challenge in this task.

Table 3. Results on using different ratios of training data (RMSE) (the best score is in bold; the score that is better than the ConvMF baseline is in underline)

Model	20%	40%	60%	80%
AMF [10]	**0.9096**	**0.875**	**0.8534**	**0.8359**
ConvMF [6]	0.9477	0.8949	0.8734	0.8578
BMF(robeta-bert)	0.9183	0.8838	0.8674	0.8516

The effect of review text lengths We investigate the effect of review text length in the MovieLens-1M. We first analyze the lengths of review texts in the MovieLens-1M data. This dataset contains 10,076 reviews, in which the length of a review is in range from 13 to 1,276 words. We calculate the ratios of text lengths (the number of words in each review) in the entire review texts in the data. There are 36.22% of review texts with less than 100 words, and 31.71% of reviews texts of which the lengths are from 100 to 200 words. The statistics show that there is a large portion of review texts of which the lengths are in the range of less than 200 words. We evaluate the performance of our BMF and the baseline AMF models on such different text lengths to investigate whether the lengths of review texts affect the behavior of our model.

Table 4. Comparison of our model and the baseline AMF model on different ranges of review text lengths (RMSE) (l: the length of review texts;the best scores are in bold)

Model	$l < 100$	$100 < l < 200$
AMF [10]	0.9251	0.9284
BMF(robeta-bert)	**0.9135**	**0.9191**

We present the results in Table 4 to compare our BMF model and the AMF model [10] in the two different ranges of text lengths: less than 100 words, and from 100 to 200 words. The results show that our BMF model outperforms the AMF model in both cases. It confirms that our BMF model is better than the AMF model in this setting, in which the review text lengths should be less than 200 words. The reason may come with the text lengths used in the pre-trained BERT model, in which using this range of text lengths may be more suitable to leverage the strength of the pre-trained BERT model. We will conduct more experiments and analyses regarding this setting of text lengths so that we can further take advantages of the power of the pre-trained BERT model in our recommender systems.

4 Conclusion

In this work we investigate the effect of utilizing pre-trained BERT models to represent item reviews to enhance matrix factorization-based recommender systems especially in sparse data settings. Instead of using conventional pre-trained

word embeddings as some previous models, we utilize pre-trained BERT for item review representations. We conducted experiments on the Movielens dataset. Although experimental results show that our model is still needed to be further investigated to improve the baseline model, we also achieve some promising findings from the intensive analyses. Our model can improve the baseline model with a specific review text lengths (less than 200 words). We plan to improve this ongoing work by conducting other analyses as well as making further modifications in both model architectures and experiment settings in future work.

References

1. Dahdouh, K., Dakkak, A., Oughdir, L., Ibriz, A.: Large-scale e-learning recommender system based on spark and hadoop. Journal of Big Data **6**(1), 1–23 (2019)
2. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
3. Felfernig, A., Le, V.M., Popescu, A., Uta, M., Tran, T.N.T., Atas, M.: An overview of recommender systems and machine learning in feature modeling and configuration. In: 15th International Working Conference on Variability Modelling of Software-Intensive Systems. pp. 1–8 (2021)
4. Feng, C., Liang, J., Song, P., Wang, Z.: A fusion collaborative filtering method for sparse data in recommender systems. Information Sciences **521**, 365–379 (2020)
5. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. In: Proceedings of the 26th international conference on world wide web. pp. 173–182 (2017)
6. Kim, D.H., Park, C., Oh, J., Lee, S., Yu, H.: Convolutional matrix factorization for document context-aware recommendation. In: Sen, S., Geyer, W., Freyne, J., Castells, P. (eds.) RecSys. pp. 233–240. ACM (2016)
7. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **42**(8), 30–37 (2009)
8. Lika, B., Kolomvatsos, K., Hadjiefthymiades, S.: Facing the cold start problem in recommender systems. Expert Systems with Applications **41**(4), 2065–2073 (2014)
9. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
10. Nguyen, H.X., Nguyen, M.L.: Attention mechanism for recommender systems. In: Proceedings of the 33rd Pacific Asia Conference on Language, Information and Computation. Association for Computational Linguistics, Japan (2019)
11. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of EMNLP. pp. 1532–1543 (2014)
12. Rendle, S., Krichene, W., Zhang, L., Anderson, J.: Neural collaborative filtering vs. matrix factorization revisited. In: Fourteenth ACM Conference on Recommender Systems. pp. 240–248 (2020)
13. Salakhutdinov, R., Mnih, A.: Probabilistic matrix factorization. In: Advances in Neural Information Processing Systems. vol. 20 (2008)
14. Wang, C., Blei, D.M.: Collaborative topic modeling for recommending scientific articles. In: Apté, C., Ghosh, J., Smyth, P. (eds.) KDD. pp. 448–456. ACM (2011)
15. Wang, H., Wang, N., Yeung, D.Y.: Collaborative deep learning for recommender systems. In: Cao, L., Zhang, C., Joachims, T., Webb, G.I., Margineantu, D.D., Williams, G. (eds.) KDD. pp. 1235–1244. ACM (2015)