Attention mechanism for recommender systems

First Author
Nguyen Huy Xuan
nguyenhx@jaist.ac.jp

Second Author
Le Minh Nguyen
nguyenml@jaist.ac.jp

Abstract

Sparseness of user item rating data affects the quality of recommender systems. To solve this problem, many approaches have been proposed. They added supplemental information to increase the accuracy. We propose a recommendation model namely attention matrix factorization (AMF) that integrates attention mechanism of the both item reviews document and item genre information into probabilistic matrix factorization (PMF). Consequently, AMF attends features which are mentioned in item reviews document and further increases the rating prediction accuracy by adding item genre information. Our experiments on the Movielens and Amazon instant video datasets show that AMF outperforms the previous traditional recommendation systems. This reveals that our model can capture subtle features of item reviews although the rating data is sparse.

1 Introduction

The sparseness of item rating is still a challenge for recommender systems. Eventually, this problem affects the rating prediction accuracy of traditional collaborative filtering (CF) approaches (Adomavicius and Tuzhilin, 2005; Herlocker et al., 2004). Recently, to improve the accuracy, several methods are proposed such as Latent Dirichlet Allocation (LDA) and Stacked Denoising AutoEncoder (SDAE). These approaches added item description information such as reviews, abstracts or synopes (Ling et al., 2014; McAuley and Leskovec, 2013; Wang and Blei, 2011; Wang et al., 2015).

Wang et al. have proposed collaborative topic regression (CTR) method which unites Latent Dirichlet Allocation (LDA) and collaborative filtering (CF) in a probabilistic approach (Wang and Blei, 2011). The author also proposed collaborative deep learning (CDL) which integrates Stacked Denoising AutoEncoder (SDAE) into probabilistic matrix factorization (PMF) (Salakhutdinov and Mnih, 2008; Wang et al., 2015). Variants of CTR were integrated with topic modeling (LDA) into collaborative filtering to analyze item description with different approaches (Ling et al., 2014; McAuley and Leskovec, 2013). However, the integrated models do not fully capture document information.

In order to overcome the issue, Donghyun Kim et al. have proposed ConvMF (Kim et al., 2016) which uses item reviews document in CNN model and further enhances the rating prediction accuracy. However, it does not mention another information such as item genre information. It also does not capture attended features of item reviews document.

The most recently, the combination between deep learning methods with CF and content-based filtering methods is also proposed. Yu Liu et al. have proposed a novel deep hybrid recommender system framework based on auto-encoders (DHA-RS) by integrating user and item side information to construct a hybrid recommender system and enhance performance (Liu et al., 2018). The author has proposed two models based on the DHA-RS framework which integrates user and item side information. Libo Zhang et al. have proposed a model combining a CF algorithm with deep learning technology (Zhang et al., 2018). This approach uses a fea-

33rd Pacific Asia Conference on Language, Information and Computation (PACLIC 33), pages 174-181, Hakodate, Japan, September 13-15, 2019

ture representation method based on a quadric polynomial regression model, which obtains the latent features more accurately by improving upon the traditional matrix factorization algorithm. These latent features are regarded as the input data of the deep neural network model, which is the second part of the proposed model and is used to predict the rating scores.

In this paper, we propose attention matrix factorization (AMF) model which integrates attention mechanism into probabilistic matrix factorization. Our model is different from previous approaches. We use attention mechanism which uses the both item reviews and item genre information to enhance rating prediction accuracy and attend features which are mentioned in item reviews information. For example, we have item reviews document and item genre as follows.

Item reviews: He ***license*** *to* ***kill*** *bond race to russia in search of the* ***stolen*** *access code.*

Item Genre[1]*: GoldenEye (1995)::****Action, Adventure, Thriller***

By adding item genre: *Action, Adventure, Thriller*, our AMF model captures attended features such as *license, kill, stolen* which are mentioned from item reviews document. Our contributions are summarized as follows.

- We propose an attention matrix factorization model which exploits ratings, item reviews documents and item genre information.
- We extensively demonstrate that AMF is a combination of PMF with attention mechanism on three datasets with more effective features representation.
- We conduct different experiments and show that AMF can facilitate the data sparsity problem in CF.

The rest of the paper is described as follows. Section 2 reviews preliminaries on the CF technique and attention neural network. Section 3 describes the AMF model and optimization method. Experimental results and evaluation AMF are presented in Section 4. Finally, we present our conclusion in Section 5.

2 Our baseline

In this section, we shortly describe the most common CF technique that is Matrix Factorization (MF) and attention network.

2.1 Matrix Factorization

Matrix factorization is one of the most popular methods in CF (Koren et al., 2009). Generally, MF model can learn low-rank representations (i.e., latent factors) of users and items in the user-item matrix, which are further used to predict new ratings between users and items. Assume that: N is a set of users; M is a set of items, and R is a rating matrix of users for items ($R \in \mathbb{R}^{N \times M}$). MF discovers the k-dimensional models, which is 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: $r_{ij} \approx \hat{r}_{ij} = {u_i}^T v_j$. The loss function $\mathcal{L}$ is calculated by equation as below.

$$\begin{aligned}\mathcal{L} = &\sum_{i}^{N}\sum_{j}^{M} f_{ij}(r_{ij} - {u_i}^T v_j)^2 + \lambda_u \sum_{i}^{N} \parallel u_i \parallel^2 \\ &+ \lambda_v \sum_{j}^{M} \parallel v_j \parallel^2\end{aligned} \tag{1}$$

Where $f_{ij} = 1$ if user u_i rated v_j; otherwise, $f_{ij} = 0$

2.2 Attention neural network

Parikh et al., proposed decomposable attention model for Natural Language Inference (Parikh et al., 2016). Inputs are two phrases represented as a sequence of word embedding vectors $a = (a_1, ..., a_{la})$ and $b = (b_1, ..., b_{lb})$. The goal of attention model is to estimate a probability that two phrases are in entailment or contradiction to each other. The core model architecture is to compose of three steps: 1) attention for generating soft-aligned to the second sentence, 2) comparison for comparing soft-aligned sentence matrices, 3) aggregation for column-wise sum over the output of the comparison step so that we obtain a fixed-size representation of every sentence.

[1] http://www.imdb.com/

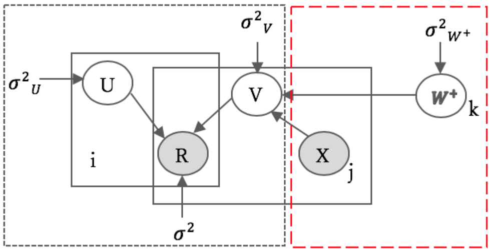

Figure 1: AMF architecture: PMF in left (dotted black); Attention neural architecture part in right (dashed red)

3 Attention mechanism for MF

In this section, we introduce our attention matrix factorization (AMF), following 3 steps: 1) We describe the probabilistic model of AMF, and introduce the main idea to combine PMF and attention mechanism in order to use ratings, item reviews documents and item genre information. 2) We describe the architecture of our attention mechanism, that generates document latent model by analyzing item reviews document and item genre. 3) We explain how to optimize our AMF.

3.1 Probabilistic Model of AMF

Our AMF is described in Figure 1, that combines an attention mechanism and PMF model. This part is cited from previous research in (Kim et al., 2016). The conditional distribution over observed ratings is given by

$$\rho(R|U,V,\sigma^2) = \prod_i^N \prod_j^M \mathcal{N}(r_{ij}|u_i{}^T v_j, \sigma^2)^{f_{ij}} \quad (2)$$

$\mathcal{N}(x|\mu,\sigma^2)$ is the Gaussian normal distribution with mean μ and variance σ^2, and f_{ij} is described in Section 2.1. The item latent model is given below.

$$v_j = att^+(W^+, X_i) + \epsilon_j \quad (3)$$

$$\epsilon_j = \mathcal{N}(0, \sigma^2{}_V f)) \quad (4)$$

Where $att^+()$ represents the output of attention architecture; X_i representing the document of item i and epsilon variable as Gaussian noise. For each weight $w_k{}^+$ in W^+, we set zero-mean spherical Gaussian prior.

$$\rho(W^+|\sigma^2{}_{W^+}) = \prod_k \mathcal{N}(w_k{}^+|0,\sigma^2{}_{W^+}) \quad (5)$$

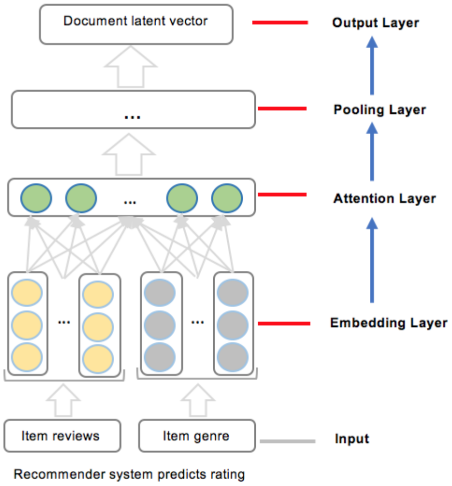

Figure 2: Our Attention neural architecture for AMF

$$\rho(V|W^+, X\sigma^2{}_V) = \prod_j^M \mathcal{N}(v_j|att^+(W^+, X_j), \sigma^2{}_V f) \quad (6)$$

where X is the set of item reviews.

3.2 Attention mechanism of AMF

In this paper, our attention mechanism uses item reviews and item genre information. Figure 2 introduces our attention architecture that consists of 4 layers described as follows.

Input of our model is both items reviews document of user for item, and item genre information.

1) Embedding Layer.

This layer is to convert a raw document into a vector. For example, we have a document with number of words is l, then we can concatenate a vector of each word into a matrix in accordance with the sequence of words. The word vectors are initialized with pre-trained word embedding model such as Glove (Pennington et al., 2014). Then, the document matrix $D \in \mathbb{R}^{q\times l}$ can be visualized as follow:

$$\begin{bmatrix} w_{11} & \cdots & w_{1i} & \cdots & w_{1l} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ w_{q1} & \cdots & w_{qi} & \cdots & w_{ql} \end{bmatrix} \quad (7)$$

in which q is the dimension of word embedding and $w[1:q,i]$ represents raw word i in the document.

2) Attention Layer.

Our attention layer is cited from previous reseach in (Parikh et al., 2016). Let $a = (a_1, ..., a_{la})$ and

$b = (b_1, ..., b_{lb})$ are the two inputs of item review and item genre with length l_a and l_b, respectively. Each $a_i, b_j \in \mathbb{R}^d$ is a word embedding vector of dimension d. Our attention mechanism is followed by three steps below.

a) Attend.

We first obtain unnormalized attention weights e_{ij}, computed by a function F', which decomposes as:

$$e_{ij} := F'(\bar{a_i}, \bar{b_j}) := F(\bar{a_i})^T F(\bar{b_j}). \qquad (8)$$

Where $\bar{a} := a$ and $\bar{b} := b$. We take F to be a feed-forward neural network with ReLU activation function (Glorot et al., 2011). These attention weights are normalized as follows:

$$\beta_i := \sum_{j=1}^{lb} \frac{exp(e_{ij})}{\sum_{k=1}^{lb} exp(e_{ik})} \bar{b_j}, \qquad (9)$$

$$\alpha_j := \sum_{i=1}^{la} \frac{exp(e_{ij})}{\sum_{k=1}^{la} exp(e_{kj})} \bar{a_i}, \qquad (10)$$

β_i is the subphrase in $\bar{b}$ that is (softly) aligned to $\bar{a_i}$ and vice versa for α_j.

b) Compare.

Next, we separately compare the aligned phrases $\{(\bar{a_i}), \beta_i\}_{i=1}^{la}$ and $\{(\bar{b_j}), \alpha_i\}_{j=1}^{lb}$ using a function G.

$$v_{1,i} := G([\bar{a_i}, \beta_i]); \forall i \in [1, ..., la], \qquad (11)$$

$$v_{2,j} := G([\bar{b_j}, \alpha_i]); \forall j \in [1, ..., lb]. \qquad (12)$$

where the brackets [.,.] denote concatenation. Thus G can jointly take into account both $\bar{a_i}$, and βi.

c) Aggregate.

Finally, we now have two sets of comparison vectors $\{v_{1,i}\}_{i=1}^{la}$ and $\{v_{2,j}\}_{j=1}^{lb}$. We first aggregate over each set by summation:

$$v_1 = \sum_{i=1}^{la} v_{1,i}; v_2 = \sum_{j=1}^{lb} v_{2,j}. \qquad (13)$$

and feed the result through a final classifier H, that is a feed forward network followed by a linear layer:

$$\hat{y} = H([v_1, v_2]), \qquad (14)$$

where $\hat{y} \in \mathbb{R}^C$represents the predicted (unnormalized) scores for each class and consequently the predicted class is given by $\hat{y}$ = argmax$x_i \hat{y}_i$.

3) Pooling Layer.

The pooling layer extracts representative features from the attention layer, and also deals with variable lengths of documents via pooling operation that constructs a fixed-length feature vector. After the attention layer, a document is represented as n_c contextual feature vectors. However, such representation has two problems: 1) there are some contextual features might not help enhance the performance, 2) the length of contextual feature vectors varies, which makes it difficult to construct the following layers. Therefore, we utilize max-pooling, which reduces the representation of a document into a n_c fixed-length vector by extracting only the maximum contextual feature from each contextual feature vector as follows.

$$d_f = [max(c^1), max(c^2), \cdots, max(c^j), \cdots, max(c^{n_c})] \qquad (15)$$

where c^j is a contextual feature vector of length $l - ws + 1$ extracted by jth shared weight W_c^j.

4) Output Layer.

From output layer, the high-level features are extracted. A document latent vector is generated by equation as below.

$$s = tanh(W_{f_2}\{tanh(W_{f_1} d_f + b_{f_1}\} + b_{f_2}) \qquad (16)$$

where W_{f_1} is projection matrices ($W_{f_1} \in \mathbb{R}^{f \times f}$); b_{f_1} and b_{f_2} are a bias vector of W_{f_1}, W_{f_2} with $s \in \mathbb{R}^k$ ($b_{f_1} \in \mathbb{R}^f, b_{f_2} \in \mathbb{R}^k$). Our Attention architecture becomes a function that exports a document latent vectors s_j of item j:

$$s_j = att^+(W^+, X_j) \qquad (17)$$

where W^+ denotes all the weight and bias variables; and X_j denotes a raw document of item j.

3.3 Optimization Methodology

Our optimization is based on previous research in (Kim et al., 2016). We utilize maximum a posteriori estimation to optimize the variables of attention.

The optimization function $\mathcal{L}$ is given below.

$$\mathcal{L}(U,V,W^+) = \sum_i^N \sum_j^M \frac{f_{ij}}{2}(r_{ij} - {u_i}^T v_j)_2$$
$$+ \frac{\lambda_U}{2}\sum_i^N \parallel u_i \parallel_2 + \frac{\lambda_V}{2}\sum_j^M \parallel v_j - att^+(W^+, X_j) \parallel_2$$
$$+ \frac{\lambda_{W^+}}{2}\sum_k^{|{w_k}^+|} \parallel {w_k}^+ \parallel_2 \tag{18}$$

where $\lambda_U = \sigma^2/{\sigma^2}_U$, $\lambda_V = \sigma^2/{\sigma^2}_V$, and $\lambda_{W^+} = \sigma^2/{\sigma^2}_{W^+}$.

The optimal solution of U (or V) is given by equations below.

$$u_i \leftarrow (VI_iV^T + \lambda_U I_K)^{-1} VR_i \tag{19}$$

$$v_i \leftarrow (UI_jU^T + \lambda_V I_K)^{-1}(UR_j + \lambda_V att^+(W, X_j)) \tag{20}$$

where I_i is a diagonal matrix with $I_{ij}, j = 1, ..., M$ and R_i is a vector with $(rij)_{j=1}^M$ for user i. For item j, I_j and R_j are similarly defined as I_i and R_i, respectively.

L is interpreted as a squared error function with L_2 regularized terms as follows.

$$\varepsilon(W^+) = \frac{\lambda_V}{2}\sum_j^M \parallel v_j - att^+(W^+, X_j) \parallel^2 +$$
$$\frac{\lambda_{W^+}}{2}\sum_k^{|{w_k}^+|} \parallel {w_k}^+ \parallel^2 + const \tag{21}$$

The back propagation algorithm is used to optimize W^+. Finally, the prediction of unknown ratings of users on items is given by equation below.

$$\begin{aligned} r_{ij} &= \mathbb{E}[r_{ij}|{u_i}^T v_j, \sigma^2] = {u_i}^T v_j \\ &= {u_i}^T(att^+(W^+, X_j) + \epsilon_j) \end{aligned} \tag{22}$$

Recall that $v_j = att^+(W^+, X_j) + \epsilon_j$

4 Experiment

In this part, we evaluate our AMF and compare with four start-of-the-art algorithms.

4.1 Experimental Setting

1) Datasets.

To evaluate rating prediction of our models, we used the MovieLens datasets[2] (ML) and Amazon Instant Video[3] (AIV). Each dataset contains user's ratings on items. Each rating value is 1-5. AIV dataset has item reviews and item descriptions. For ML data, we obtained item reviews of corresponding items from imdb site[4]. For the genre information, we extract from the item files (*_movies.dat) (i.e., $itemID$ :: $itemtitle$ :: $genre1|genre2|genre3|$...).

We also pre-processed item reviews documents for all datasets similar to previous approaches (Wang and Blei, 2011; Wang et al., 2015). We removed users and items that have less than 3 ratings and do not have their description documents. Table 1 shows the statistics of each dataset. We see that even when several users are removed by preprocessing, AIV is still sparse compared with the ML dataset.

2) Baselines.

We compared our AMF model with two previous methods, which are PMF (Salakhutdinov and Mnih, 2008), CTR (Wang and Blei, 2011) as well as two deep learning methods, which are CDL (Wang et al., 2015) and ConvMF (Kim et al., 2016).

3) Evaluation Metrics.

To evaluate the performance of each model, we randomly divided each dataset into three sets: 10% for test, 10% for validation and 80% for training. The training set contains at least one ratings on each user and item so that PMF deals with all users and items. Since our purpose is to conduct rating prediction, we use root mean squared error (RMSE) as the evaluation metrics.

$$RMSE = \sqrt{\frac{\sum_{i,j}^{N,M}(r_{ij} - \hat{r}_{ij})^2}{\#\ of\ ratings}} \tag{23}$$

4) Parameter Settings.

We set the training data with different percentage (20%, 40%, 80%). For the latent dimension of U and V, we set 50 according to previous work in (Wang et al., 2015) and initialized U, V randomly

[2] https://grouplens.org/datasets/movielens/

[3] http://jmcauley.ucsd.edu/data/amazon/

[4] http://www.imdb.com/

Dataset	**Item information**	**Genre information**	**# Users**	**# Items**	**# Ratings**	**Density**
ML-1m	Item reviews	Item genre	6,040	3,544	993,482	4.641%
ML-10m	Item reviews	Item genre	69,878	10,073	9,945,875	1.413%
AIV	Item reviews	Item genre	29,757	15,149	135,188	0.030%

Table 1: Data statistic on three real-world datasets

	ML-1m		ML-10m		AIV	
Model	λ_U	λ_V	λ_U	λ_V	λ_U	λ_V
PMF	0.01	10000	10	100	0.1	0.1
CTR	100	1	10	100	10	0.1
CDL	10	100	100	10	0.1	100
ConvMF	100	10	10	100	1	100
AMF	**10**	**60**	**10**	**60**	**1**	**60**

Table 2: Parameter Setting of λ_U and λ_V

Model	ML-1m	ML-10m	AIV
PMF	0.8961	0.8312	1.412
CTR	0.8968	0.8276	1.552
CDL	0.8876	0.8176	1.3694
ConvMF	0.8578	0.7995	1.209
AMF	**0.8359**	**0.7834**	**1.106**
Improvement	2.19%	1.61%	10.3%

Table 3: RMSE

from 0 to 1. The best performance values of parameters λ_U, λ_V of each model are described in Table 2.

4.2 Experimental Results

1) Evaluate Results.

Table 3 evaluates rating prediction error of our AMF model and four competitors. Note that "Improvement" shows the relative improvements of AMF over the best competitor. AMF achieves better performance than ConvMF, CDL, CTR, PMF. Specifically, our AMF has strong effectiveness on sparse dataset that is AIV data.

With MovieLens, the improvements of AMF over the best competitor, ConvMF, are 2.19% on ML-1m and 1.61% on ML-10m.

With AIV data, the improvement of AMF over the best competitor, ConvMF, is 10.3%.

2) Evaluate Results Over Sparseness Datasets. We set the different sparsenesses by randomly sampling with ML-1m, ML-10m and AIV datasets. Table 4 shows AMF still has robust and good performance when compared with the best competitor (ConvMF). This implies the effectiveness of incorporating item genre information in attention mechanism. Specifically, we observe that the improvements of AMF over ConvMF are 2.81% on ML-1m and 2.35% on ML-10m and 14.81% on AIV when training set is only 20%.

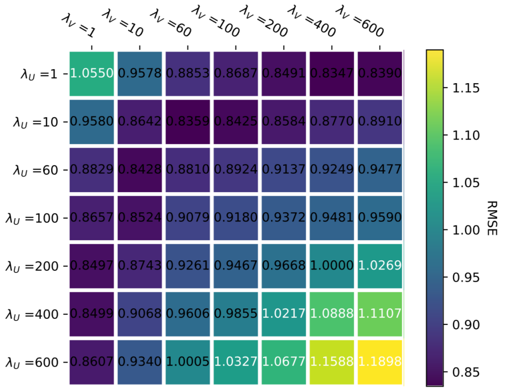

Figure 3: Parameter analysis of λ_U and λ_V on ML-1m dataset

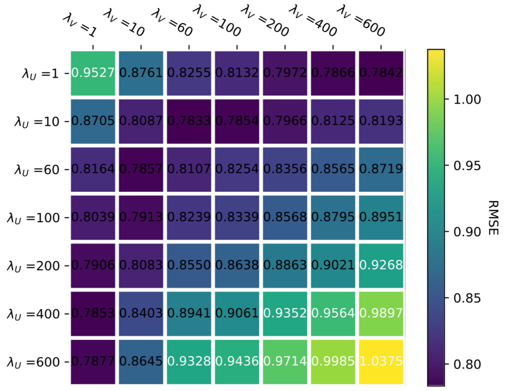

Figure 4: Parameter analysis of λ_U and λ_V on ML-10m dataset

3) Impact of Parameters.

Figure 3, 4, 5 show the impact of λ_U and λ_V for three datasets ML-1m, ML-10m and AIV. We see

	ML-1m			ML-10m			AIV		
model	20%	40%	80%	20%	40%	80%	20%	40%	80%
ConvMF	0.9477	0.8949	0.8578	0.8896	0.8515	0.7995	1.4426	1.3584	1.2090
AMF	**0.9196**	**0.8755**	**0.8359**	**0.8661**	**0.8255**	**0.7834**	**1.2945**	**1.2171**	**1.1060**
Improvement	2.81%	1.94%	2.19%	2.35%	2.6%	1.61%	14.81%	14.13%	10.30%

Table 4: RMSE over sparseness of datasets

Model	Using Information	ML-1m	ML-10m	AIV
ConvMF	Item reviews	0.8578	0.7995	1.209
Concatenation	Item reviews + Item genre with concatenation	0.8513	0.8161	1.1891
AMF	**Item reviews + Item genre with attention**	**0.8359**	**0.7834**	**1.106**

Table 5: Comparing RMSE between AMF, Concatenation and ConvMF

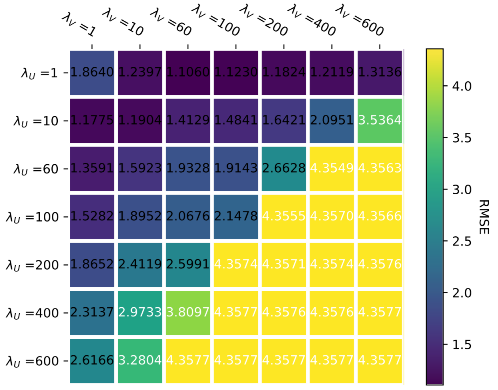

Figure 5: Parameter analysis of λ_U and λ_V on AIV dataset

that when rating data becomes sparse, λ_U and λ_V decrease to produce the best results. In fact, the values of (λ_U, λ_V) of AMF are (10, 60), (10, 60) and (1, 60) on ML-1m, ML-10m and AIV, respectively.

4) Impact of Attention

We analyze the effectiveness of attention mechanism to document latent vector which improve rating prediction accuracy. We compare with another implementation that is *concatenation* between item reviews and item genre information.

In Table 5, our AMF model still has better performance than concatenation method. The results also show that concatenation method has better performance than ConvMF on ML-1m and AIV datasets. Specifically, we observe that the improvements of AMF over concatenation method are 1.54% on ML-1m and 3.27% on ML-10m. In the case AIV, it has strong effectiveness with 8.31% of improvement.

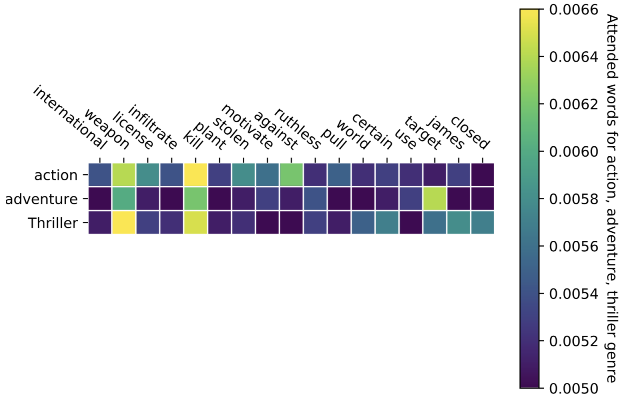

Figure 6: Attended feature of text comments

Figure 6 is our case study, we figure out the output of our model using attention mechanism with item genre information. The highlight points are the features attended by item genre information. These features have strong effectiveness in improving rating prediction accuracy. Moreover, they also help us to understand reviews document for items easily.

In Figure 6, we observed as follows.

- When item genre is **action**, the words **weapon**, **kill**, **against** are attended.
- When item genre is **adventure**, the words **weapon**, **kill**, **target** are attended.
- When item genre is **thriller**, the words **weapon**, **kill** are attended.

5 Conclusion

In this paper, we proposed AMF model that combines attention mechanism into PMF to enhance the rating prediction accuracy. Extensive results demonstrate that attention mechanism of AMF significantly outperforms the other competitors, which implies that AMF deals with the data sparsity problem by adding item genre information. Moreover, our model can figure out attended features for item reviews document which make us understand which information is attended from item reviews document.

References

Gediminas Adomavicius and Alexander Tuzhilin. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep sparse rectifier neural networks. In Geoffrey J. Gordon, David B. Dunson, and Miroslav Dudk, editors, *AISTATS*, volume 15 of *JMLR Proceedings*, pages 315–323. JMLR.org.

J.L. Herlocker, J.A. Konstan, L.G. Terveen, and J.T. Riedl. 2004. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1):5–53.

Dong Hyun Kim, Chanyoung Park, Jinoh Oh, Sungyoung Lee, and Hwanjo Yu. 2016. Convolutional matrix factorization for document context-aware recommendation. In Shilad Sen, Werner Geyer, Jill Freyne, and Pablo Castells, editors, *RecSys*, pages 233–240. ACM.

Y. Koren, R. Bell, and C. Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37.

Guang Ling, Michael R. Lyu, and Irwin King. 2014. Ratings meet reviews, a combined approach to recommend. In Alfred Kobsa, Michelle X. Zhou, Martin Ester, and Yehuda Koren, editors, *RecSys*, pages 105–112. ACM.

Y. Liu, S. Wang, M. S. Khan, and J. He. 2018. A novel deep hybrid recommender system based on autoencoder with neural collaborative filtering. *Big Data Mining and Analytics*, 1(3):211–221, Sep.

Julian J. McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: understanding rating dimensions with review text. In Qiang Yang, Irwin King, Qing Li, Pearl Pu, and George Karypis, editors, *RecSys*, pages 165–172. ACM.

Ankur P. Parikh, Oscar Tckstrm, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. *CoRR*, abs/1606.01933.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

Ruslan Salakhutdinov and Andriy Mnih. 2008. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, volume 20.

Chong Wang and David M. Blei. 2011. Collaborative topic modeling for recommending scientific articles. In Chid Apt, Joydeep Ghosh, and Padhraic Smyth, editors, *KDD*, pages 448–456. ACM.

Hao Wang, Naiyan Wang, and Dit-Yan Yeung. 2015. Collaborative deep learning for recommender systems. In Longbing Cao, Chengqi Zhang, Thorsten Joachims, Geoffrey I. Webb, Dragos D. Margineantu, and Graham Williams, editors, *KDD*, pages 1235–1244. ACM.

L. Zhang, T. Luo, F. Zhang, and Y. Wu. 2018. A recommendation model based on deep neural network. *IEEE Access*, 6:9454–9463.