# A Collaborative Neural Model for Rating Prediction by Leveraging User Reviews and Product Images

Wenwen Ye[1], Yongfeng Zhang[2], Wayne Xin Zhao[3⋆] Xu Chen[1] and Zheng Qin[1]

[1] School of Software, Tsinghua University
[2] College of Information and Computer Sciences, University of Massachusetts Amherst
[3] School of Information, Renmin University of China

**Abstract.** Product images and user reviews are two types of important side information to improve recommender systems. Product images capture users' appearance preference, while user reviews reflect customers' opinions on product properties that might not be directly visible. They can complement each other to jointly improve the recommendation accuracy. In this paper, we present a novel collaborative neural model for rating prediction by jointly utilizing user reviews and product images. First, product images are leveraged to enhance the item representation. Furthermore, in order to utilize user reviews, we couple the processes of rating prediction and review generation via a deep neural network. Similar to the multi-task learning, the extracted hidden features from the neural network are shared to predict the rating using the softmax function and generate the review content using LSTM-based model respectively. To our knowledge, it is the first time that both product images and user reviews are jointly utilized in a unified neural network model for rating prediction, which can combine the benefits from both kinds of information. Extensive experiments on four real-world datasets demonstrate the superiority of our proposed model over several competitive baselines.

## 1 Introduction

Nowadays, recommender systems have been widely used in various online services, such as e-commerce, news-reading and video-sharing websites. Traditional methods mainly capture the interactions between users and items, *e.g.,* factorizing the user-item rating matrix [14]. Recently, with the ever increasing of user-generated content, multiple types of side information have been utilized to improve the recommendation accuracy. Among these side information, product images and user reviews have received much research attention.

Intuitively, product images can directly reflect users' appearance preference (*e.g.,* clothing styles and phone looks), which are usually seldom (or even hard) to be described in words, while user reviews can uncover the customers' favored characters that might be invisible from the product images (*e.g.,* clothing quality and phone weight), they can complement each other for better understanding users' interests. To see this, we present an illustrative example in Figure 1. A user has assigned a high rating and posted a short review on an item: *"Very good quality! I like it."*. On one hand, from the

⋆ Corresponding author

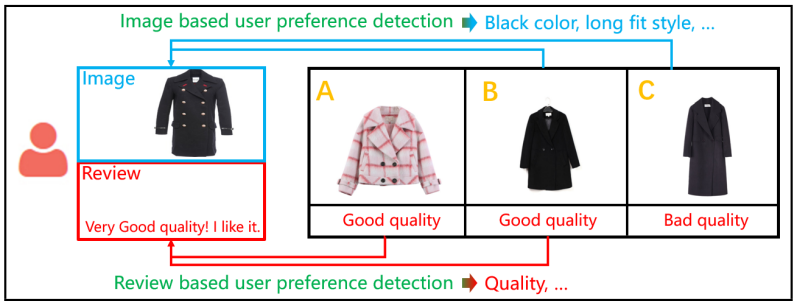



**Fig. 1.** An illustrative example on the complementary effect of visual and textual features on recommender systems.

product image, we can clearly identify the visual characteristics that the user prefers, *e.g.,* the color and the style. On the other hand, as a complementary signal, the user review can further reveal the other important aspects considered by the user, *i.e.,* quality. In this example, with the help of visual features, $B$ and $C$ could be selected as candidate recommendations to the user, since they are similar to the purchased product by her in terms of color and style. Furthermore, by considering review information, $C$ will be filtered out due to bad experiences on *clothing quality* indicated in historical reviews, which is an important factor to consider for the current user.

Existing works have demonstrated the effectiveness of using either user reviews or product images [15, 9, 17] for rating prediction. However, few studies have investigated the effect of their integration for more accurate recommendations. Hence, we would like to study two research questions: (1) whether it is possible to jointly utilize user reviews and product images in a unified model; (2) how such an integration improves the performance compared with traditional methods using a single kind of information.

A major challenge to answer the two questions is how to properly and effectively combine heterogeneous information (*i.e.,* user review and product image) together. Traditional methods [14, 20] would be less effective when faced with multiple kinds of heterogeneous information due to the limitation of simple linear structures [10]. Fortunately, the rapid development of deep learning techniques sheds light on this problem because of its superiority in the field of multi-modal fusing [19, 21, 13, 24].

In this paper, we present a novel collaborative **N**eural **R**ating **P**rediction model by jointly modeling the **T**extual features collected from user reviews and the **V**isual features extracted from product images (called **NRPTV**). We develop the model in a deep learning framework. Our model is built on the core components, *i.e.,* user and item representations, which encodes useful information from users and items. To integrate visual features, we combine the item latent factor (obtained by using a lookup layer) with the transformed visual features (derived from item images) as the image-enhanced item representation. The derived user and item representations are subsequently fed into a Multi-Layer Perceptron (MLP) as input for rating prediction. Furthermore, in order to utilize user reviews, we couple the processes of rating prediction and review generation via a MLP component. Similar to the multi-task learning, the extracted hidden features from the MLP component are shared to predict the rating using the softmax function and generate the review content using a LSTM-based model respectively. In this way,

our model can utilize both textual and visual features for recommender systems, and combine the benefits from both kinds of features.

To the best of our knowledge, it is the first time that both product images and user reviews are jointly characterized in a unified neural network model for rating prediction. Extensive experiments on four real-world datasets demonstrate the superiority of our proposed model over several competitive baselines. Our experiment results also show that using a combination of both types of features leads to a substantial performance improvement compared to that using only either type.

In the rest of the paper, we first review the related work in Section 2, and present the proposed model in Section 3. Section 4 gives the experimental results and analysis, and Section 5 concludes the paper.

## 2 Related Work

Recommender systems have attracted much attention from the research community and industry [2]. We mainly review two highly related research directions.

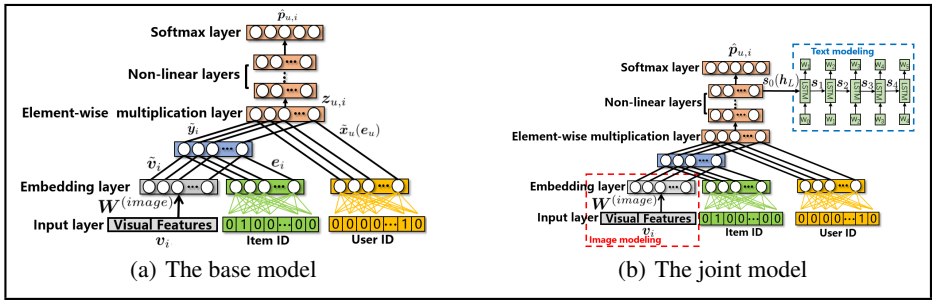
**Review-based Recommendation.** Many efforts have been made to incorporate user reviews into traditional recommendation algorithms [29]. The major assumption is that review contain important textual features which are potentially improve the recommendation performance. Typical methods include correlating review topics with the latent factors in matrix factorization [15, 25], feature-level information utilization [31, 3], and the distributed semantic modeling [28]. A problem with these studies is that they usually make the bag-of-words assumption, and sequential word order has been ignored.

**Image-based Recommendation.** In recent years, visual features have been leveraged by recommender systems [9, 17, 4, 30], which aim to capture important appearance characteristics of the items. Specially, visual features have been incorporated into the Bayesian Personalized Ranking framework [9] and sequential prediction task [6]. Furthermore, visual features have been used to better find visually complementary items to a query image [17].

Although review and image data are important and complementary to recommender systems, to our knowledge, few studies can jointly utilize both user review and product image. Our work take the initiative to develop a collaborative neural model leveraging both visual and textual features for rating prediction. We are also aware of the recent progress of deep learning techniques on recommender systems [25, 5, 26, 10, 32]. They seldom consider the incorporation of review and image. As a comparison, our focus is to borrow the benefits of deep learning models and perform effective heterogeneous data fusion and utilization for rating prediction.

## 3 A Collaborative Neural Model for Rating Prediction

In this section, we present the preliminaries and the proposed collaborative neural models for rating prediction.




(a) The base model (b) The joint model

**Fig. 2.** Our proposed base model (left) and joint model (right) for rating prediction. Boxes with dotted circles represent embedding or hidden vectors. The gray, green and orange boxes correspond to $\boldsymbol{v}_i$, $\boldsymbol{e}_u$ and $\boldsymbol{e}_i$ respectively.

### 3.1 Preliminaries

Formally, let $u$ and $i$ denote a user and an item respectively, and $r_{u,i}$ denote the rating of user $u$ on item $i$. As usual, the rating values are discrete and range from 1 to $R$, *e.g.,* the five-star rating mechanism on Amazon ($R = 5$). We assume the corresponding images for item $i$ are available, denoted by a vector $\boldsymbol{v}_i$. Each rating $r_{u,i}$ is associated with a user review, denoted by a vector $\boldsymbol{w}_{u,i}$, in which $w_{u,i,k}$ denotes the $k$-th word token in the review. Given an observed rating dataset $\mathcal{D} = \{\langle u, i, r_{u,i}, \boldsymbol{w}_{u,i}, \boldsymbol{v}_i \rangle\}$, the *rating prediction* task aims to predict the ratings of a user on the un-rated items.

A typical approach to rating prediction is to factorize the interactions between users and items by learning user and item representations. In standard matrix factorization (MF), it approximately estimates the missing entry $r_{u,i}$ by the inner product between user latent representation $\boldsymbol{x}_u$ and item latent representation $\boldsymbol{y}_i$, *i.e.,* $\hat{r}_{u,i} = \boldsymbol{x}_u^\top \cdot \boldsymbol{y}_i$. Although such a general approach is widely used, it adopts the linear matrix factorization, and may not be effective and flexible to incorporate multiple types of heterogeneous information. Inspired by the recent progress in the multi-modality deep learning, next we build our approach in a deep learning framework.

### 3.2 The Base Model: Image-enhanced Rating Prediction

Following the classic MF approach, our base model also keeps user and item representations. Furthermore, we assume that both kinds of representations not only characterize the user-item interactions, but also can encode other useful side information for rating prediction. Formally, let $\tilde{\boldsymbol{x}}_u$ and $\tilde{\boldsymbol{y}}_i$ denote the user and item representations in our model, both of which are a $K$-dimensional vector, *i.e.,* $\tilde{\boldsymbol{x}}_u, \tilde{\boldsymbol{y}}_i \in \mathbb{R}^K$. The focus of this section is to incorporate image information into the item representation and develop a neural model for rating prediction.

**User and item embeddings as representations**. First, with the help of the look-up layer, the one-hot input of either user or item IDs are first projected into low-dimensional embeddings, which are similar to the latent factors in the context of matrix factorization. Let $\boldsymbol{e}_u (\in \mathbb{R}^K)$ and $\boldsymbol{e}_i (\in \mathbb{R}^K)$ denote the user and item embeddings, we then set

the basic user and item representations as the corresponding embeddings by:

$$\tilde{\boldsymbol{x}}_u = \boldsymbol{e}_u, \tag{1}$$

$$\tilde{\boldsymbol{y}}_i = \boldsymbol{e}_i. \tag{2}$$

**Imaged-enhanced item representation**. As shown in Figure 1, product images encode appearance characteristics which are potentially useful to improve recommender systems. Now, we study how to incorporate product images in our model. Recall that each item $i$ is associated with a visual feature vector, denoted by $\boldsymbol{v}_i$. To set $\boldsymbol{v}_i$, we use a pre-trained approach to generate visual features from raw product images using the deep learning framework of CAFFE [12]. Following [9], we adopt the CAFFE reference model with five convolutional layers followed by three fully-connected layers that has been pre-trained on 1.2 million IMAGENET images. For item $i$, the second fully-connected layer is taken as the visual feature vector $\boldsymbol{v}_i$, which is a feature vector of length 4096. To combine $\boldsymbol{v}_i$ with $\boldsymbol{e}_i$, we map $\boldsymbol{v}_i$ into a $K$-dimensional vector $\tilde{\boldsymbol{v}}_i$ using a linear transformation, *i.e.,* $\tilde{\boldsymbol{v}}_i = \boldsymbol{W}^{(image)} \cdot \boldsymbol{v}_i$, where $\boldsymbol{W}^{(image)} \in \mathbb{R}^{K \times 4096}$ is the transformation matrix for images. Once $\tilde{\boldsymbol{v}}_i$ and $\boldsymbol{e}_i$ have the same dimensionality, we further preform the element-wise vector product to derive the new item representation,

$$\tilde{\boldsymbol{y}}_i = \tilde{\boldsymbol{v}}_i \odot \boldsymbol{e}_i. \tag{3}$$

Note that there can be alternative ways to to combine $\tilde{\boldsymbol{v}}_i$ with $\boldsymbol{e}_i$, *i.e.,* the vector concatenation. In our experiments, our current choice leads to a good performance and we adopt it as the combination method.

**Rating prediction through a MLP classifier**. Recall that our rating values have totally $R$ choices. Hence, we adopt a classification approach to solve the rating prediction task, which has been shown to be effective in [33, 23]. More specially, each rating value $r_{u,i}$ is considered as a class label, and will be represented as a $R$-dimensional one-hot vector, denoted by $\boldsymbol{p}_{u,i}$, where only the $r_{u,i}$-th entry is equal to 1. We implement the classification model using a Multi-Layer Perceptron (MLP) with $L$ hidden layers. Our input consists of both user and item representations, namely $\tilde{\boldsymbol{x}}_u$ and $\tilde{\boldsymbol{y}}_i$. We follow the similar way in Eq. 3 to combine two representation into a single input feature vector $\boldsymbol{z}_{u,i} \in \mathbb{R}^K$

$$\boldsymbol{z}_{u,i} = \tilde{\boldsymbol{x}}_u \odot \tilde{\boldsymbol{y}}_i, \tag{4}$$

where $\tilde{\boldsymbol{x}}_u$ and $\tilde{\boldsymbol{y}}_i$ are defined in Eq. 1 and 3 respectively. Furthermore, let $\boldsymbol{h}_l$ denote the corresponding output of the $l$-th hidden layer, which is derived on top of the $(l-1)$-th hidden layer for $l \in \{1, 2, ...L\}$

$$\boldsymbol{h}_l = f(\boldsymbol{h}_{l-1}), \tag{5}$$

where $f(\cdot)$ is an non-linear activation function implemented by the Rectifier Linear Unit (ReLU) in our model because ReLU is usually more resistable to overfitting and works well in practice [7]. To feed the input of the MLP classifier, we set $\boldsymbol{h}_0 = \boldsymbol{z}_{u,i}$. Finally, the last softmax layer takes in $\boldsymbol{h}_L$ to produce a probability distribution for $R$ classes,

denoted by $\hat{\boldsymbol{p}}_{u,i}$. The loss function computes the cross-entropy loss between the ground truth (*i.e.,* $\boldsymbol{p}_{u,i}$) and the predicted results (*i.e.,* $\hat{\boldsymbol{p}}_{u,i}$):

$$\begin{aligned}\mathcal{L}_{base} &= \sum_{\langle u,i\rangle \in \mathcal{D}} \text{CrossEnt}(\boldsymbol{p}_{u,i}, \hat{\boldsymbol{p}}_{u,i}), \\ &= \sum_{\langle u,i\rangle \in \mathcal{D}} \sum_{r=1}^{R} -p_{u,i,r} \cdot \log \hat{p}_{u,i,r}.\end{aligned} \tag{6}$$

We present a model sketch in Fig 2(a). As we can see, the model has incorporated the visual features in the item representation. The combined user and item representations are fed into a MLP classifier with $L$ non-linear layers. We adopt the non-linear transformation since visual features may not be directly ready in a form for rating prediction. Deep neural models could be effective to transform the visual information into a better representation for rating prediction.

### 3.3 The Joint Model: Integrating User Review with Product Image for Rating Prediction

In the above, visual features have been fused into rating prediction model via the improved item representation. Now, we study how to integrate user reviews into the base model. Following Section 3.2, we can take a similar approach to incorporate review information, in which the textual features would be used as input to enhance either the user or item representations. However, such a straightforward approach may be practically infeasible because: (1) given a user, her review information on some product may not be available for rating prediction model, since the reviewing behavior usually occur after purchase behavior; and (2) user reviews and product images represent two kinds of heterogeneous side information, it is likely to perform poorly by simply fusing two kinds of information.

**Overview of the model**. To address this difficulty, our idea is to treat the review content as another kind of output signal besides the ratings. We do not modify the bottom neural architecture for user and item representations in the base model. Instead, we take the output of the last non-linear layer (*i.e.,* $\boldsymbol{h}_L$) in the MLP component, and generate review contents (*i.e.,* $\boldsymbol{w}_{u,i}$) based on it. Since $\boldsymbol{h}_L$ was previously passed into a softmax layer for rating prediction, our current approach couples the two processes: rating prediction and review generation. In essence, the idea is similar to the multi-task or multi-modality learning [1]. Following this idea, the next problem is how to model the review generation.

**LSTM-based review modeling**. Most of the existing review-based recommendation models make the bag-of-words assumption [15, 22], and they ignore the effect of sequential word order on the semantics of review text. Consider two sample reviews: "*The screen is good, while the battery is unsatisfactory*" and "*The screen is unsatisfactory, while the battery is good*". Although they consist of the same words, they convey totally different semantics. Hence, we have to consider the sequential order in

review generation. To capture the word sequential information, we adopt the long short term memory (LSTM) [11] network, which has been successfully applied to a number of sequence text modeling tasks [24, 27]. Formally, let $V$ be the vocabulary size, and $\boldsymbol{w}_{u,i} = \{w_{u,i,0}, ..., w_{u,i,k}, ..., w_{u,i,n_{u,i}-1}\}$ denote the review published by user $u$ on item $i$, where $n_{u,i}$ is the review length and $w_{u,i,k}$ is the $(k+1)$-th token in the review. The LSTM model generates the review content in a sequential way as follows:

$$\boldsymbol{s}_k = \text{LSTM}(\boldsymbol{s}_{k-1}, w_{u,i,k-1}, \Phi), \tag{7}$$

$$p(w_{u,i,k}|\boldsymbol{w}_{u,i,<k}, \Phi) = \text{softmax}_{w_{u,i,k}}(\boldsymbol{s}_k, \Phi), \tag{8}$$

where $1 \leq k \leq n_{u,i} - 1$, $\boldsymbol{w}_{u,i,<k}$ denote the preceding $k$ words, the $\boldsymbol{s}_k$ is the state vector for the $k$-th step (*i.e.,* the $k$-th word), LSTM$(\cdot)$ is the standard LSTM unit [11], softmax$(\cdot)$ is the softmax function which transforms the hidden state into a $V$-dimensional word generation probability distributions, and $\Phi$ denote the set of all the necessary parameters for text generation.

**The joint optimization model**. The above formulation presents the review text generation independent from the rating prediction. Next, we couple these two parts via the shared hidden layer in the MLP component. We set the initial state vector to the last hidden layer in the MLP as below

$$\boldsymbol{s}_0 = \boldsymbol{h}_L, \tag{9}$$

where $\boldsymbol{h}_L$ is defined by Eq. 5 in Section 3.2 and the subscript of "0" indicates the zero state of LSTM. Finally, the overall loss function is given below

$$\begin{aligned} \mathcal{L}_{joint} =& \alpha \sum_{\langle u,i \rangle \in \mathcal{D}} \sum_{k=1}^{n_{u,i}-1} -\log p(w_{u,i,k}|\boldsymbol{w}_{u,i,<k}, \Phi) \\ &+ (1-\alpha) \sum_{\langle u,i \rangle \in \mathcal{D}} \text{CrossEnt}(\boldsymbol{p}_{u,i}, \hat{\boldsymbol{p}}_{u,i}), \end{aligned} \tag{10}$$

where CrossEnt$(\boldsymbol{p}_{u,i}, \hat{\boldsymbol{p}}_{u,i})$ is the cross-entropy defined in Eq. 6, $p(w_{u,i,k}|\boldsymbol{w}_{u,i,<k}; \Phi)$ is the word generation probability defined in Eq. 8, and $\alpha$ $(0 \leq \alpha < 1)$ is a tuning parameter that balances the effects of the two parts. When $\alpha = 0$, the model becomes the base model in Section 3.2. In our joint model, the visual features are integrated via the item representation, and the textual features are integrated in the output signals. Note that although we describe the two parts separately, our approach links both parts in a unified optimization model. All the parameters can be jointly learned by optimizing the loss function Eq. 10 using the stochastic gradient descent (SGD) method.

We denote the model by **NRPTV**, *i.e.,* a collaborative **N**eural **R**ating **P**rediction model based on **T**extual features and **V**isual features. We present the overview of the final model in Fig. 2(b). By comparing Fig. 2(a) and Fig. 2(b), we can see that the textual features have been integrated into the base model in a similar way as multi-task learning. The purpose of the MLP component is to enhance the capacity to integrate heterogeneous information and improve the prediction performance.

# 4 Experiments

In this section, we present the experiments. We begin by introducing the experimental setup, and then report and analyze the experimental results.

## 4.1 Datasets.

We use four Amazon datasets shared in [16], which are from four diverse product categories. The statistics of these datasets are shown in Table 1.

**Table 1.** Basic statistics of the datasets.

| Datasets | #Users | #Items | #Ratings | $\frac{\#Reviews}{\#Users}$ | Density |
|---|---|---|---|---|---|
| Music | 1,492 | 900 | 7,931 | 5.55 | 0.59% |
| Patio | 1,686 | 962 | 11,740 | 6.96 | 0.72% |
| Auto | 2,928 | 1,835 | 18,308 | 6.25 | 0.34% |
| Clothing | 39,387 | 23,033 | 278,677 | 7.07 | 0.03% |

**Methods to compare.** To demonstrate the effectiveness of our model, we consider the following methods for performance comparison:

- **PMF** [18]: PMF model represents the classic rating prediction approach, which only utilizes the user-item rating matrix.
- **HFT** [15]: HFT model aligns topics from topic models with latent factors in matrix factorization. It is a commonly used review-based rating prediction baseline.
- **modified VBPR(mVBPR)** [9]: VBPR is a pioneering work which incorporates product images into top-$N$ recommendation. To adapt VBPR to rating prediction, we modify the original ranking loss to the least square loss.
- **modified NeuMF(mNeuMF)** [10]: NeuMF is a recently proposed neural network model for top-$N$ recommendation. To adapt NeuMF to rating prediction, we modify the original ranking loss to the cross-entropy loss as our model.
- **NRPTV**: NRPTV is our proposed model which jointly utilizes visual and textual information.

## 4.2 Parameter settings

Our models are implemented using the library TENSORFLOW. The model parameters are randomly initialized according to the uniform distribution. The learning rate of SGD is determined by grid searching in the set $\{1, 0.1, 0.01, 0.001, 0.0001\}$. We set the number of non-linear layers in the MLP component to 3, and the dimensionality are set to $\{40, 20, 5\}$ to form a tower structure [8]. The tuning parameter $\alpha$ is first set to 0.1 empirically, and will be analyzed in detail in the following experiment. For fair comparison, we set all the biases as 0 in the baseline models. In our experiments, we randomly split the full dataset into training and test sets with a split ratio of 7:3. To evaluate the performance of the comparison methods, RMSE (Root of the Mean Square Error) is adopted as the evaluation metric.

**Results and Analysis.** In this section, we present the experimental results and analysis on the task of rating prediction. The RMSE results of different methods are reported in Table 2. From Table 2, we can make the following observations:

**Table 2.** Performance comparison of different methods using RMSE (smaller is better). $K$ denotes the dimensionality.

| Datasets | $K$ | PMF | mNeuMF | mVBPR | HFT | NRPTV |
|---|---|---|---|---|---|---|
| Music | 50 | 1.076 | 1.075 | 1.069 | 1.067 | **1.063** |
| | 100 | 1.075 | 1.072 | 1.066 | 1.064 | **1.061** |
| | 150 | 1.077 | 1.074 | 1.071 | 1.068 | **1.064** |
| | 200 | 1.078 | 1.072 | 1.068 | 1.067 | **1.066** |
| Patio | 50 | 1.242 | 1.240 | 1.237 | 1.216 | **1.211** |
| | 100 | 1.236 | 1.234 | 1.231 | 1.215 | **1.209** |
| | 150 | 1.244 | 1.241 | 1.239 | 1.224 | **1.213** |
| | 200 | 1.245 | 1.241 | 1.238 | 1.227 | **1.216** |
| Auto | 50 | 1.179 | 1.174 | 1.171 | 1.169 | **1.159** |
| | 100 | 1.172 | 1.171 | 1.170 | 1.168 | **1.157** |
| | 150 | 1.183 | 1.181 | 1.173 | 1.171 | **1.163** |
| | 200 | 1.189 | 1.184 | 1.176 | 1.173 | **1.167** |
| Clothing | 50 | 1.390 | 1.386 | 1.381 | 1.380 | **1.374** |
| | 100 | 1.389 | 1.382 | 1.377 | 1.379 | **1.373** |
| | 150 | 1.394 | 1.387 | 1.382 | 1.385 | **1.376** |
| | 200 | 1.399 | 1.392 | 1.382 | 1.388 | **1.380** |

– Overall, a dimensionality of 100 works well for all the methods. A smaller dimensionality may not be able to achieve powerful predictability, while a larger dimensionality tends to overfit on the training data.
– mNeuMF is better than PMF because it can incorporate more powerful non-linear transformation, which may lead to a better performance.
– With additional side information, either image or review, both mVBPR ($+image$) and HFT ($+review$) further improve substantially over mNeuMF and PMF in most cases. This finding indicates that side information is important to consider in rating prediction.
– Our proposed model NRPTV ($+image+review$) is consistently better than all the baselines on the four datasets with four different dimensionalities. This is because: NRPTV jointly utilizes images and reviews, and it also combines the benefits from deep learning.

### 4.3 Detailed Analysis of Our Model

In the above, we have shown the effectiveness of our model NRPTV. Now, we carry out more detailed analysis for NRPTV in order to analyze the individual effect of different components or parameters on the performance. At each time, we only check one component or parameter, while the rest will be fixed to the optimal settings. In what

follows, we fix the dimensionality (*i.e.,* $K$) as 100, since Table 2 has shown that the dimensionality of 100 gives good performance.

**Influence of the tuning parameter $\alpha$.** An important parameter to tune is $\alpha$ in Eq. 10. We vary it from 0.1 to 0.9 with a gap of 0.1. We present the tuning results for $\alpha$ in Fig. 3. On the *Music* dataset, the performance achieves the best when $\alpha = 0.4$, while on the *Auto* dataset, the performance achieves the best when $\alpha = 0.6$. Due to space limit, we only report the results on the two datasets. For other datasets, we have also found that $\alpha \in [0.4, 0.6]$ usually gives the best performance. $\alpha$ controls the importance of the review generation component when adding to the base model. The observations indicate we should make a suitable balance, neither too large nor too small, between the base model and added review model component.

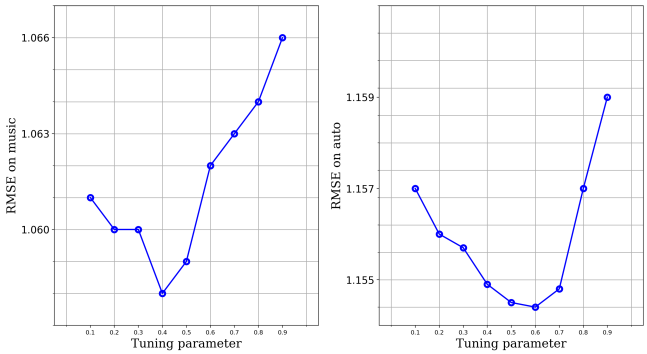



**Fig. 3.** The influence of tuning parameter $\alpha$.

**Influence of the image and review modeling components.** As shown in Fig. 2(b), our model jointly utilizes both the image and review in a unified model. We now examine how each component affects the prediction performance. To examine it, we implement two variants based on the full model, which remove either the review or image components respectively, called *NRPT* ($+text$) and *NRPV* ($+image$). Then, we compare the performance of NRPTV, NRPT and NRPV, and report the RMSE results on four datasets in Fig. 4. It can be observed that NRPTV is consistently better than both NRPT and NRPV, which indicates that both components are important to rating prediction. Another interesting finding is that the two variants NRPT and NRPV alteratively perform better than each other. For example, on the *Clothing* dataset, visual features seem to play an more important role to improve the performance, while on the *Auto* dataset, textual features contribute more to the final performance. The finding is actually quite intuitive. Visual features are more powerful to capture appearance characteristics such as color and style. As a comparison, textual features are more powerful to capture the usage experiences such as easy to use and convenience.

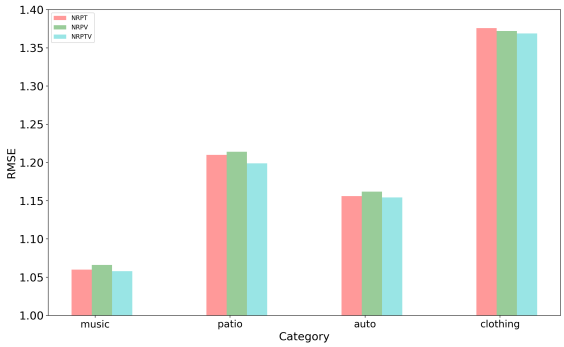



**Fig. 4.** Performance comparison among NRPT, NRPV and NRPTV based on the RMSE results. NRPT includes the image modeling component, NRPV includes the review modeling component, and NRPTV includes both components.

## 5 Conclusion

In this paper, we proposed a novel collaborative neural model for rating prediction by jointly modeling user reviews and product images. Our work is motivated by the intuition that visual and textual features can complement each other to improve recommendation accuracy. Extensive experiments demonstrate the effectiveness of our proposed model and the importance to combine both kinds of side information. Our work has made the attempt to characterize heterogenous side information using deep neural models in recommender systems. As future work, we will consider developing a general model which can integrate more kinds of side information. We will also study how to improve recommendation interpretability using these side information.

## Acknowledgment

Xin Zhao was partially supported by the National Natural Science Foundation of China under grant 61502502 and the Beijing Natural Science Foundation under grant 4162032.

## References

1. Argyriou, A., Evgeniou, T., Pontil, M.: Multi-task feature learning. In: NIPS (2007)
2. Bobadilla, J., Ortega, F., Hernando, A., Gutiérrez, A.: Recommender systems survey. Knowledge-based systems (2013)
3. Chen, X., Qin, Z., Zhang, Y., Xu, T.: Learning to rank features for recommendation over multiple categories. In: SIGIR (2016)
4. Chen, X., Zhang, Y., Ai, Q., Xu, H., Yan, J., Qin, Z.: Personalized key frame recommendation. In: SIGIR (2017)
5. Cheng, H.T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., Anderson, G., Corrado, G., Chai, W., Ispir, M., et al.: Wide & deep learning for recommender systems. In: Recsys Workshop on DLRS (2016)
6. Cui, Q., Wu, S., Liu, Q., Wang, L.: A visual and textual recurrent neural network for sequential prediction. arXiv preprint arXiv:1611.06668 (2016)
7. Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: Aistats (2011)

8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
9. He, R., McAuley, J.: Vbpr: Visual bayesian personalized ranking from implicit feedback. In: AAAI (2016)
10. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. In: WWW (2017)
11. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation (1997)
12. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. In: MM (2014)
13. Kiros, R., Salakhutdinov, R., Zemel, R.S.: Multimodal neural language models. In: ICML (2014)
14. Koren, Y., Bell, R., Volinsky, C., et al.: Matrix factorization techniques for recommender systems. Computer (2009)
15. McAuley, J., Leskovec, J.: Hidden factors and hidden topics: understanding rating dimensions with review text. In: Recsys (2013)
16. McAuley, J., Pandey, R., Leskovec, J.: Inferring networks of substitutable and complementary products. In: KDD (2015)
17. McAuley, J., Targett, C., Shi, Q., Van Den Hengel, A.: Image-based recommendations on styles and substitutes. In: SIGIR (2015)
18. Mnih, A., Salakhutdinov, R.: Probabilistic matrix factorization. In: NIPS (2007)
19. Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., Ng, A.Y.: Multimodal deep learning. In: ICML (2011)
20. Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: Bpr: Bayesian personalized ranking from implicit feedback. In: UAI (2009)
21. Srivastava, N., Salakhutdinov, R.R.: Multimodal learning with deep boltzmann machines. In: NIPS (2012)
22. Tan, Y., Zhang, M., Liu, Y., Ma, S.: Rating-boosted latent topics: Understanding users and items with ratings and reviews. In: IJCAI (2016)
23. Tang, D., Qin, B., Liu, T., Yang, Y.: User modeling with neural network for review rating prediction. In: IJCAI (2015)
24. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: A neural image caption generator. In: CVPR (2015)
25. Wang, H., Wang, N., Yeung, D.Y.: Collaborative deep learning for recommender systems. In: SIGKDD (2015)
26. Wang, H., Xingjian, S., Yeung, D.Y.: Collaborative recurrent autoencoder: Recommend while learning to fill in the blanks. In: NIPS (2016)
27. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A.C., Salakhutdinov, R., Zemel, R.S., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. In: ICML (2015)
28. Zhang, W., Yuan, Q., Han, J., Wang, J.: Collaborative multi-level embedding learning from reviews for rating prediction. In: IJCAI (2016)
29. Zhang, Y.: Explainable recommendation: Theory and applications. arXiv preprint arXiv:1708.06409 (2017)
30. Zhang, Y., Ai, Q., Chen, X., Croft, W.: Joint representation learning for top-n recommendation with heterogeneous information sources. In: CIKM (2017)
31. Zhang, Y., Lai, G., Zhang, M., Zhang, Y., Liu, Y., Ma, S.: Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In: SIGIR (2014)
32. Zhao, W.X., Li, S., He, Y., Chang, E.Y., Wen, J.R., Li, X.: Connecting social media to e-commerce: Cold-start product recommendation using microblogging information. TKDE (2016)
33. Zheng, Y., Tang, B., Ding, W., Zhou, H.: A neural autoregressive approach to collaborative filtering. In: ICML (2016)