Table 1: Statistics of the datasets. *Musical Instruments* is abbreviated as Music.

Datasets	#User	#Item	#Interaction	Density
Music	1429	900	10245	0.797%
Automotive	2928	1835	20441	0.380%
Toy	19412	11924	167472	0.072%
Yelp	34547	47010	1523939	0.094%

• **Yelp**[6]: This is a large-scale dataset including users' rating and review behaviors for different restaurants. Because the raw data is very large, we pre-process it by removing the users and items with less than 20 ratings. The statistics of these datasets can be seen in Table 1.

Evaluation method and baselines Root Mean Square Error (RMSE) is leveraged in our experiments to evaluate different models. Suppose the predicted and real ratings from u to v are $\hat{r}_{uv}$ and r_{uv}, respectively. The RMSE score is calculated by:

$$\text{RMSE} = \sqrt{\frac{1}{|\mathcal{T}|} \sum_{(u,v)\in\mathcal{T}} (r_{uv} - \hat{r}_{uv})^2}, \qquad (12)$$

where $\mathcal{T}$ is the set of user-item pairs in the testing set, and lower RMSE score means better performance. In our experiments, the following representative models are selected as the baselines:

- **PMF:** This is a traditional matrix factorization method (Mnih and Salakhutdinov 2008), and the model parameters are learned by stochastic gradient decent (SGD).
- **GRU4Rec:** This is a well known sequential recommender model (Hidasi et al. 2016), where each previously interacted item is accordingly fed into each time step.
- **Time-LSTM** This is a time-aware sequential recommender method, where the time interval information (Zhu et al. 2017) is incorporated in the modeling process.
- **Time-LSTM++:** This method is an advanced version of Time-LSTM, where the input of each step not only contains item ID, but also includes review and rating information as in equation 7 for more comprehensive user profiling.
- **NARRE:** This is a state-of-the-art explainable recommendation method (Chen et al. 2018a), which has been verified to outperform many promising algorithms including NMF, SVD++, HFT and DeepCoNN on Amazon and Yelp datasets. We implemented it based on the authors' public code[7].

Implementation details For each user behavior sequence, the last and second last interactions are used for testing and validation, while the other interactions are left for training. In our model, the batch size as well as the learning rate are determined in the range of $\{50, 100, 150\}$ and $\{0.001, 0.01, 0.1, 1\}$, respectively. The user/item embedding size K is tuned in the range of $\{8, 16, 32, 64, 128\}$, and we will discuss its influence on the model performance in the following sections. For the user review information, we first pre-process it based on the Stanford Core NLP tool[8], and then the word embeddings are pre-trained based on the Skip-gram model[9]. The baselines designed for Top-N recommendation are revised to optimize the RMSE score.

Table 2: The results of comparing our model with the baselines in terms of RMSE.

Dataset	Music	Automotive	Toy	Yelp
PMF	1.0706	1.0106	1.1226	1.3411
GRU4Rec	1.0111	0.9723	1.0363	1.3011
Time-LSTM	0.9901	0.9615	0.9963	1.2821
Time-LSTM++	0.9878	0.9435	0.9805	1.2711
NARRE	0.9784	0.9199	0.9690	1.2507
DER	**0.9678**	**0.8981**	**0.9535**	**1.2314**

Evaluation on Rating Prediction

Overall performance From the results shown in Table 2, we can see: the simple PMF method performed worst because it fails to capture the sequential properties for user behavior modeling, and also cannot borrow the power of review information to enhance the user/item representations. Time-LSTM and Time-LSTM++ performed better than GRU4Rec, which is consistent with the previous study (Zhu et al. 2017), and verifies the effectiveness of time interval information for user dynamic preference modeling. NARRE outperformed Time-LSTM++, and the reason can be that, for a target item, NARRE utilizes all its review information to provide informative signals to assist the rating prediction process, and the attention mechanism further make it powerful to discriminatively enhance the impact of the valuable review information, while reducing the noise influence. Encouragingly, DER consistently performed better than the best baseline NARRE on all the datasets. Comparing with NARRE, which represents each user as a static embedding, the carefully designed T-GRU architecture enables us to accurately model user dynamic preference, which facilitates more adaptive and reasonable user profiling, and eventually leads to improved rating prediction.

Influence of the embedding size K. In this section, we investigate how the embedding size influences our model's performance, and due to the space limitation, unless specified, we only report the results on the Automotive dataset. We observe the performance changes by tuning the embedding size K in the range of $\{8, 16, 32, 64, 128\}$. From the result presented in Figure 3, we can see: our model achieved the best performances when the embedding size was relative small (*i.e.*, $K = 8$) , while larger K didn't help to further improve the results. This observation actually agrees with many previous studies (Li et al. 2016; Zhang et al. 2017;

[6] https://www.kaggle.com/yelp-dataset/yelp-dataset/data

[7] https://github.com/THUIR/NARRE

[8] https://stanfordnlp.github.io/CoreNLP/

[9] http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/