Effective Visual Clustering for Personalized Multimodal Fashion Recommendation

Gyuseok Lee
Carnegie Mellon University
Pittsburgh, PA 15213
gyuseokl@andrew.cmu.edu

Bryan Lim
Carnegie Mellon University
Pittsburgh, PA 15213
hyunkukl@andrew.cmu.edu

Dong-Hwan Jang
Carnegie Mellon University
Pittsburgh, PA 15213
donghwan@andrew.cmu.edu

Joohwan Ko
Carnegie Mellon University
Pittsburgh, PA 15213
joohwank@andrew.cmu.edu

Abstract

In this project, we present an effective visual clustering method for personalized multimodal fashion recommendations. After using multimodal feature extractions from the Amazon fashion dataset, we applied K-means clustering for item embedding to solve the cold start problem and noise embedding problem in the recommender system. Our experiment has exceeded the performance of the baseline as well.

1 Introduction

Traditional recommendation models use no more than a single variable as an input. However, as many state-of-the-art multimodal models have been proposed, there are many efforts of integrating multiple input variables into the recommender system. One of the well-known recommendation systems is personalized fashion recommendation. Thanks to its early adaptation in the industry and a good amount of datasets, many researchers tried to bring up different types of models including multimodal attention networks. Here, in this project, we would like to build an efficient multimodal attention network that improves some of the drawbacks of previous models. Therefore, by using the visual and text modality, we recommend the fashion to the user and try to solve the cold start problem when a new user or item is added. Furthermore, we would like to try to design a new object function, so we intend to make our model perform better.

2 Literature Review

Our project is mainly inspired by the work by [1]. The authors present a multimodal attention network for personalized fashion recommendations in this paper. It proposes a model named VECF(Visually Explainable Collaborative Filtering). This filtering model comprises two parts: 1. fine-grained visual preference modeling and 2. review enhanced model supervision. By using CNN models to extract features from fashion images, the filter applies an attention network to find out feature maps for given images. Then, it adds embedding vectors generated from review(text) data. Other works by [5] present a learning model which learns to separate feature embeddings of different modality inputs (text, image, user information, etc.) into chunks where the correlation between chunks is close to 0. Also, the work by [6] proposes a model of separately learned encoders for the user and item information.

Final Report.

3 Approach, Dataset, and Evaluation

3.1 Problem Formulation

Referring to the objective function in the paper([1]), we will use the **likeness score** in the model from user i to item j is predicted as below:

$$\hat{y}_{ij} = P\left(\boldsymbol{p}_i, \boldsymbol{q}_j \odot \left(\boldsymbol{W}_I \boldsymbol{I}_{ij}\right)\right). \tag{1}$$

Then the final objective function to be optimized is as follows(where you may refer to the original work for the specific notation):

$$\begin{aligned}\mathcal{L} = & \sum_{i\in\mathcal{U}} \left(\sum_{j\in\mathcal{V}_{+}^{i}} \log \sigma\left(\hat{y}_{ij}\right) + \sum_{j\in\mathcal{V}/\mathcal{V}_{+}^{i}} \log\left(1-\sigma\left(\hat{y}_{ij}\right)\right) \right) \\ & + \beta \sum_{(i,j)\in O} \sum_{t=1}^{l_{ij}} \log p\left(w_{ij}^{t} \mid w_{ij}^{1:t-1}, I_{ij}^{t-1}\right) - \lambda\|\Theta\|_{2}^{2}.\end{aligned} \tag{2}$$

In this project, we will be using this objective function for our model.

Though the above objective function can handle the item that has been trained before, it cannot handle the new item because of the cold start problem for the item id. Since the previous work used element-wise multiplication to combine the user and item embedding vectors as in Equation (1), the new item embedding vector assigned with 0 or average value can greatly affect the final prediction score.

For real-world applications, new item addition is inevitable. If the new item id is not trained since the recommendation score is too low for every user at the beginning, the new item will not be recommended to any user. Therefore we will introduce the k-means clustering algorithm in the next section and how we tried to overcome this issue.

3.2 Improvement on the Model Design

Visual Feature Extraction We will use 1) better CNN model for the visual feature extraction and 2) better convolutional block attention module for the attention network. In the original paper, the authors use the last layer of the VGG16 model [8] to extract the visual feature of the item. However, the VGG16 model is not the best model for the visual feature extraction. It is proven that depending on the backbone model, the performance of the overall model can be greatly improved. Therefore, we will adopt the ResNet50 model as our new backbone model for the visual feature extraction and compare the performance with the original model in the ablation study.

Review Feature Extraction Previous method uses the LSTM model[4] to predict the review sentence embedding. However, the LSTM model is well-known to have the vanishing gradient problem, and not suitable for the long-term dependency which could be critical for the review sentence prediction. If the text modailty is not well handled, it can even lead to the performance degradation of the model. To solve this problem, we adopt the Transformer-based model, Sentence-BERT [7]. Sentence-BERT is a pre-trained model that can be used for the sentence embedding generation. First, we will use the pre-trained model to generate the sentence embedding for the review data. Then, we will use the sentence embedding as the pseudo label to train the feature extraction model. Since Sentence-BERT is a pre-trained model, we expect that the model can handle wild text data much better than the LSTM model which is trained from scratch in the original paper. The feature extracted from the Sentence-BERT model has 768 dimensions, and we will use MLP network to predict the review embedding from visual feature.

K-Means Clustering As mentioned in the previous section, there are always issues of cold-start problems with new users in our recommender system. There could be many way of solving this issue but we used an algorithm called K-means clustering to resolve this issue. We added the K-means clustering algorithm for the new item id assignment based on the visual feature of the item. First,

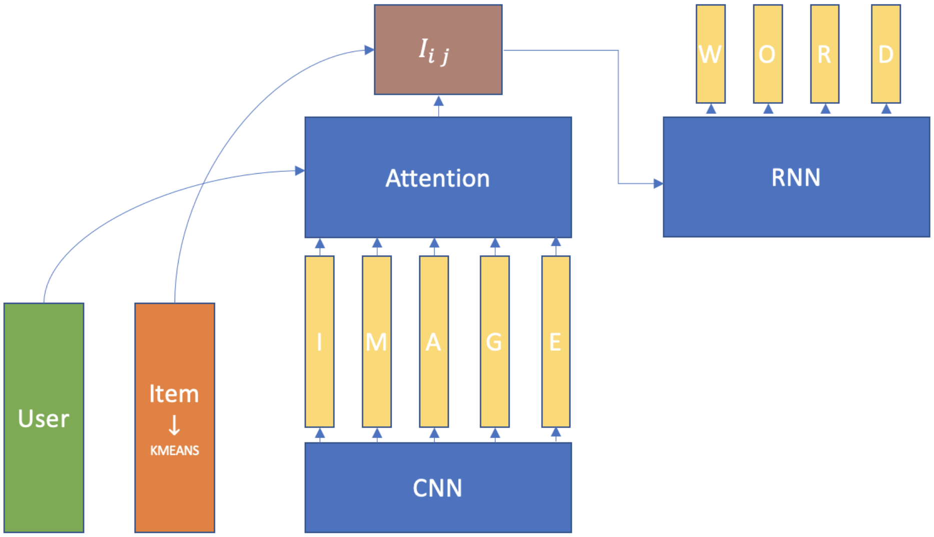

Figure 1: Overall Model Architecture

we will extract the visual feature of the item using the last layer of the pre-trained ResNet-50 model. Then, we will use the K-means clustering algorithm to cluster the visual feature of the item into K clusters. Finally, the new item id will be assigned to the cluster that has the closest visual feature to the new item in the test. With the predicted item id, we can use the trained model to predict the recommendation score for the new item.

3.3 Experimental Setup

Dataset There are a bunch of benchmark datasets we can use: FashionCV dataset([9]), Amazon dataset([2]), and the Tradesy.com dataset([3]). For this project, we will be using the Amazon dataset. And as the dataset is too big to test a newly introduced algorithm, we only chose one category for our train which is *Baby Clothes* in the Amazon. There are 6992 users and 7818 items in the data. In order to make settings robust to cold start problems, we applied negative sampling for train dataset so that we are also training for the cases that we are having new users.

Model As we already mentioned, our overall model structure follows [1]. The only difference is that the item embedding has been replaced by the center value of the cluster of K-means. The overall description of the model is as follows. Firstly, CNN Model(i.e., VGG Net) extracts feature map for item images like 14 x 14 x 512 . for each image of i, we can flatten the feature map and consequently get image matrix $F_j \in R^{512\times196}$. After that, we can make $I_{i,j}$, where each of i,j means user and item id, by calculating attention weights like:

$$I_{i,j} = F_j\alpha_{ij} = \sum_{k=1}^{h} \alpha_{ijk} f_j^k \quad (3)$$

$$\alpha_{ijk} = W_2[ReLU(W_1[W_u p_i \odot W_f f_j^k])] \quad (4)$$

$$\alpha_{ijk} = \frac{exp(\alpha_{ijk})}{\sum_{k'=1}^{h} exp(\alpha_{ijk'})} \quad (5)$$

Specifically, after obtaining the attention score using the relationship between image and user embedding, $I_{i,j}$ is created. This has the premise that the user's preference can be sufficiently derived from the correlation with the given image. Then, in the form of Equation (1), the score between the user and the item is obtained. At this time, h = 196, VGG Network was used as CNN, and LSTM was used as RNN. LSTM is used for the next word prediction task, and the review dataset written by the user in the corresponding item was used. Through this, we attempted to utilize weak supervision and multi-modality.

Evaluation Metric For the evaluation metric, we will follow the metric in [1] as well, which is a well-known evaluation method in the field of recommender systems. We will split the dataset into 80% training and 20% test. After training the model, for each user, the model ranks all the items and truncates the ranking list at position N to evaluate the Top-N recommendation problem(they chose 10 for N). For the measure, we will use F1, Hit Ratio(HR), and Normalized Discounted Cumulative Gain(NDCG) for different models.

$$NDCG_{@K} = \frac{DCG_{@K}}{IDCG_{@K}}$$

$$IDCG_{@K} = \sum_{i=1}^{K^{\text{ideal}}} \frac{G_i^{\text{ideal}}}{\log_2(i+1)}$$

Furthermore, compared to the original paper, we will check the performance of the model on the cold start problem. By spliting the dataset in item level, we will input the new item to the baseline model and our model to see the performance difference. We expect that our model can handle the cold start problem by assigning the new item to the cluster that has the closest visual feature to the new item in test.

Computing Resources We have used various GPUs(Nvidia Tesla T4, V100, and 3060 ti) for feature extraction and training. As we have set our codebase on the Amazon Web Service(AWS), other miscellaneous computings were done on AWS.

4 Experiment Results

Table 1 shows the result for our experiments on different clusters: 1000, 2500, and 5000. The results are based on 5 users with the most interactions in the dataset. And as we can see in the experiment results, we can conclude that K-cluster with 2500 clusters performs the best among these three.

Table 1: Best performance of 1000, 2500, and 5000 clusters experiments

	NDCG@10 (%)	NDCG@50 (%)	NDCG@100 (%)
No Clusters (Baseline)	2.13	2.55	2.55
K-cluster 1000	4.93	4.93	6.03
K-cluster 2500	**6.86**	**6.19**	**6.69**
K-cluster 5000		3.29	3.29

Due to the limitations of computational resources, we could not have done experiments on the different number of clusters. However, we still believe that the optimal number of clusters would be near 2500 and also utilizing K-means clustering does increase the performance of the algorithm even in the case of cold start problems.

5 Future Works

There are some future works that we wanted to do but could not because of time limitations. First, we want to try different attention modules. We would use the convolutional block attention module (CBAM) [10] for the attention network. Though cross-modality attention is used in the original paper, it tends to only focus on the one subregion in 14x14 feature map due to the softmax function in the attention network. It greatly hinders the performance and interpretability of the model because the customer can have an interest in the different subregions of the item. Therefore, we will use the CBAM module to improve the performance and interpretability of the model. CBAM is a block attention module that utilizes channel-wise attention and spatial attention. With channel-wise attention, the model can focus on important channels that customers are interested in. Also, with spatial attention, we can find the region of interest based on the sigmoid function, which allows the model to focus on the different subregions of the item. We will adopt the CBAM module at the end of the visual feature extraction part and compare the performance with the previous attention module.

Also, more experiments could be done in the future. We will perform the ablation study to see the performance difference of the model with different components: 1) different backbone model for the visual feature extraction, 2) different attention module for the attention network, and 3) different feature extraction model for the review feature extraction.

6 Conclusion

We made a multimodal attention network for personalized recommendations. By combining visual and review feature extraction for the dataset, we made a recommender system that came for fashion shopping. Also, we have mainly focused our contribution on solving the cold start problem. By using K-means clustering, we made it possible to assign a cluster to new coming users in the system. And with various experiments, we could conclude that using K-means clustering does help increase the performance of the model, and it is also efficient in solving cold start problems and noise embedding problem.

7 Team Members Work Distribution

We did equally contribute to this project, and each one of us has a different background in this field. Joohwan Ko and Donghwan Jang focused on developing the multimodal part by integrating various state-of-the-art techniques. Gyuseok Lee and Bryan Lim focused on the recommender system part. Also for the final presentation and report, Donghwan and Bryan did most of the part in the presentation and Joohwan and Gyuseok did most of the work in writing the final report.

Acknowledgments

We thank professor Alex Hauptmann and Rita Singh for giving valuable comments on each of our weekly presentation.

References

[1] X. Chen, H. Chen, H. Xu, Y. Zhang, Y. Cao, Z. Qin, and H. Zha. Personalized fashion recommendation with visual explanations based on multimodal attention network: Towards visually explainable recommendation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 765–774, 2019.

[2] R. He and J. McAuley. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *proceedings of the 25th international conference on world wide web*, pages 507–517, 2016.

[3] R. He and J. McAuley. Vbpr: visual bayesian personalized ranking from implicit feedback. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

[4] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[5] F. Liu, Z. Cheng, H. Chen, A. Liu, L. Nie, and M. Kankanhalli. Disentangled multimodal representation learning for recommendation. *arXiv preprint arXiv:2203.05406*, 2022.

[6] S. Luo, X. Lu, J. Wu, and J. Yuan. Aware neural recommendation with cross-modality mutual attention. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 3293–3297, 2021.

[7] N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[8] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[9] X. Song, F. Feng, J. Liu, Z. Li, L. Nie, and J. Ma. Neurostylist: Neural compatibility modeling for clothing matching. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 753–761, 2017.

[10] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.