Sefamerve Research at The SIGIR eCom'22: Outfit Recommendation Based on Collaborative Filtering

Selman Delil*
Sefamerve R&D Center
Istanbul, Turkey
selman.delil@sefamerve.com
selmandelil@gmail.com

Birol Kuyumcu*
Sefamerve R&D Center
Istanbul, Turkey
birol.kuyumcu@sefamerve.com
birolkuyumcu@gmail.com

ABSTRACT

This paper describes our contribution to SIGIR eCom'22: Fashion Outfits Challenge. Given the real outfits produced by stylists and fashion experts, the task is to predict the most suitable missing item of a real outfit, out of a list of candidate products. Our approach considers the missing-item-prediction task as a neighborhood-based collaborative filtering problem. We experimented with several approaches both item and user-based and reached the best prediction score of 0.68 which is ranked 3rd in the competition.

KEYWORDS

fashion outfits challenge, collaborative filtering, neighborhood-based collaborative filtering

ACM Reference Format:
Selman Delil and Birol Kuyumcu. 2018. Sefamerve Research at The SIGIR eCom'22: Outfit Recommendation Based on Collaborative Filtering. In *Proceedings of The 2022 SIGIR Workshop On eCommerce (SIGIR eCom'22).* ACM, New York, NY, USA, 3 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

The main goal of the Fashion Outfit Challenge is to develop a model that can generate outfits for each individual product. The organizers formed the competition by giving the real outfits produced by stylists and fashion experts and asked to predict the most suitable missing item of a real outfit, out of a list of candidate products [1].

We studied on the assumption that the outfit challenge task may be considered as a collaborative filtering (CF) problem. The CF recommendation system is the process of filtering or evaluating items through the opinions of other people [2]. Because the outfit combinations are chosen by certain designers, we have considered each outfit as a user choice.

One of the common methods of collaborative filtering (CF) is the neighborhood-based methods. Neighborhood-based CF algorithms rely on the assumption that similar users tend to display similar behavior patterns and similar items receive similar ratings [6]. These methods are used to determine the best item recommendations for the target user, or the best user recommendations for a target product. Hence, we can talk about two main CF approaches as user-based and item-based.

2 DATASET

The dataset consists of a list of outfits described by the products that compose it, and the Farfetch—leading platform for online luxury fashion shopping—product images and metadata, and outfits composition.

- # of products : 398670
- # of outfits: 300000
- # of products in training outfits dataset: 344797
- # of products not included training outfit dataset: 53873

Outfits consist of a minimum of 3 and a maximum of 14 products. Number of products distribution in the outfit dataset

3 METHODOLOGY

3.1 User-Based CF

We have formulated our problem with the assumption that each outfit is a user in order to fit the data we have into the UB-CF problem. For this reason, we created a vector with the length of the number of products corresponding to each user (each outfit). In this vector, if the relevant product is in the outfit set, it is defined as 1, otherwise it is defined as 0. Thus, a matrix with the size of 300K x 400K was formed, where the outfits were represented by rows and items by columns. Compared to the total number of products, our matrix was formed in a Spars structure, since the number of products in each outfit set is low.

Since nearest neighbor algorithms rely on exact matches, they sacrifice the coverage and accuracy of the recommendation. Many customer matches have no correlation at all, as users must have evaluated at least two products at the same time to measure similarity between them.

We consider outfits in the test dataset as users and products as items to adapt our data to UB-CF. In order to solve this sparsity problem, the dimension reduction process has been applied on the vector representing the similarity between the users. Unsupervised methods UMAP [5] and LDA (latent-dirichlet-allocation) [3] were used to find user similars in the data set. We created a 64 length embedding vector for each user. By using products in similar outfit, the candidate product scores were calculated, and the candidate with the highest score was selected.

*Both authors contributed equally to this research.

SIGIR eCom'22, July 15, 2022, Madrid, Spain

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

Co-occurred combined-attributes	candidate combined-attribute	Frequency	Probability
Shirts-GREEN-MEN & Jackets-BLUE-UNISEX	Trainers-WHITE-MEN	4	0.50
	Clutch Bags-PURPLE-MEN	3	0.375
	Trousers-BLACK-MEN	1	0.125
	TOTAL	**8**	**1**

Table 1: Calculation of combined-attributes

Scoring is calculated according to the number of co-occurrence between the products in the incomplete outfits and the candidate products, as in the algorithm presented below.

```
FOR each candidates
    Score = 0
    FOR each product in incomplete_outfit
        Score += CoOcCnt(cand_i , incmlt_outfit_p)
    CandidateScore(Score)
```

3.2 Item-Based CF

In this study, the item-based collaborative filtering (IB-CF) model has been tested which is the second approach in the neighborhood-based CF method.

We calculated the co-occurrence of the products in the training dataset based on product_id. Accordingly, the candidate products for the incomplete_outfits were selected using the scoring calculation derived from frequency of co-occurrence with the products in the outfit dataset (Table 2). The candidate product with the highest score was selected, if the candidates received equal score or zero, the first one was accepted as a recommendation.

First Product	Second Product	Count	Probability
15379678	15360881	4	0.4
	15781925	2	0.2
	16204075	2	0.2
	16260894	1	0.1
	14817366	1	0.1

Table 2: IB-CF Model Score Calculation

3.3 Item-Based CF + Item-attribute model

The main disadvantage of the IB-CF model is that sometimes all candidates can get 0 score which is calculated based on the co-occurrence frequency of the outfit products and candidate products. Considering the 300K outfit in the training set, there are 53873 products (approximately 18%) that have never been used with any other combination before.

In this case, the best option is to make a probabilistic estimation based on the characteristics of the products instead of suggesting a random product. For this purpose, we mapped all products to the new space produced as a combination of the Category, Main color and Gender features, which we call item-attribute model. All the new co-occurrence scores calculated again association of this representative combined_attribute class. The probability values were obtained by normalizing frequencies dividing by the total number of observations Table 1.

We utilized the item-attribute model for those samples candidates get 0 or 1 scores from the main model. If the score from the first model is 1 or less, the product with the highest probability is presented as a suggestion.

3.4 UB-CF + IB-CF Hybrid Model (LightFM)

To reach a better estimation score, we suggest combining both user-based and item-based collaborative filtering models. We utilized LightFM, an effective recommendation system library which makes it possible to incorporate both item and user metadata into traditional matrix factorization algorithms. LightFM represents each user and item as the sum of hidden representations of their properties. This allows recommendations to generalize to new items (via item properties) and new users (via user properties) [4].

With LightFM, we apply a hybrid mode by providing the product attributes to the model together with the user defined outfit combinations.

The following variables are used as product attributes: [product_family, product_category, product_sub_category, product_gender, product_main_colour, product_second_color]

4 RESULTS AND DISCUSSION

As shown in the Table 3, we obtained the best test result by IB-CF + item-attribute model.Implementation of UB + IB Hybrid model provided better than UB model, but still quite far from IB-CF, and IB-CF + item-attribute model.

Model	Result (FITB)
UB-CF (LDA) [a]	0.148
UB-CF (UMAP) [b]	0.155
IB-CF [c]	0.65
IB-CF + item-attribute model [d]	0.68
UB + IB Hybrid Model (LightFM) [e]	0.515

Table 3: Model prediction results

[a] Embedding vector similarity with a proximate nearest neighborhood. Vectorizer LDA with n_components = 64, similarity search with annoy model.
[b] Embedding vector similarity with a proximate nearest neighborhood. Vectorizer UMAP with n_components = 64 similarity search with annoy model.
[c] Item based collaborative filtering. The number of outfits in which the products co-occurred were calculated as candidate scores.
[d] Winner model: Addition to IB-CF, co-occurrence scores calculated based on combined attributes for those samples candidates get 0 or 1 scores from the main IB-CF model.
[e] LightFM recommendation, UB-CF with item attributes.

As stated in the experiment design phrase, there is a possibility that more than one outfit can actually be created by the same designer/user. Since there is no information on which combinations were created by the same stylists, it is considered that the training with UB-CF did not give good results.

5 CONCLUSION

Our approach considers missing-item-prediction-task as a neighborhood-based collaborative filtering problem. We experimented with several approaches both item and user based, and reached the best prediction score as 0.68 which is ranked 3th in the competition.

Regarding the UB-CF model, since there is no data about designers in the data set, we ignored the fact that more than one outfit can actually be created by the same designer/user. This obviously contradicted our assumption especially where the evaluation metric was defined as “Fill in the Blank (FITB)” in the research design.

We couldn't use the image dataset in our model due to time restriction, but we might have a suggestion for the future problem definition phrases. For example; given incomplete outfits researchers might be asked to predict the most suitable item from a certain product category. In that case, using product images could have been more applicable for the problem.

REFERENCES

[1] Luis Baia, Eder Martins, Diogo Goncalves, Vanessa Marinho, Felipe Viegas, Ana Silva, and Tiago Otto. 2022. Farfetch Outfits Challenge on E-Commerce Workshop. In *SIGIR eCom 2022.*

[2] Jonathan L Herlocker, Joseph A Konstan, and John Riedl. 2000. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work*. 241–250.

[3] Hamed Jelodar, Yongli Wang, Chi Yuan, Xia Feng, Xiahui Jiang, Yanchao Li, and Liang Zhao. 2019. Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey. *Multimedia Tools and Applications* 78, 11 (2019), 15169–15211.

[4] Maciej Kula. 2015. Metadata embeddings for user and item cold-start recommendations. *arXiv preprint arXiv:1507.08439* (2015).

[5] Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426* (2018).

[6] Tianqing Zhu, Yongli Ren, Wanlei Zhou, Jia Rong, and Ping Xiong. 2014. An effective privacy preserving algorithm for neighborhood-based collaborative filtering. *Future Generation Computer Systems* 36 (2014), 142–155.