IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received July 4, 2019, accepted July 11, 2019, date of publication July 23, 2019, date of current version August 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2928848

Differentiated Fashion Recommendation Using Knowledge Graph and Data Augmentation

CAIRONG YAN, YIZHOU CHEN, AND LINGJIE ZHOU
School of Computer Science and Technology, Donghua University, Shanghai 201620, China

Corresponding author: Cairong Yan (cryan@dhu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61402100 and Grant 61602109, in part by the Fundamental Research Funds for the Central Universities of China under Grant 16D111210, and in part by the Natural Science Foundation of Shanghai under Grant 19ZR1401900.

ABSTRACT E-commerce recommender systems (RSs) can help users quickly find what they need or new products they might be interested in. To continuously enhance user trust in the website, improve page visits and dwell time, and most importantly, increase gross merchandise value (GMV), it is crucial to understand and capture the important information hidden in the data, which has a great impact on user choice. The fashion e-commerce websites can collect the attributes of items and users as well as the user purchase behaviors, but lack the fine-grained classification of the items and the implicit relationship between items and users. This paper focuses on Amazon fashion dataset, one of the most widely used datasets in the fashion field. A differentiated recommendation framework is proposed that provides different recommendation paths for active and inactive users to improve the overall recommendation quality. In the framework, a data augmentation algorithm based on transfer learning is proposed to filter out the irrelevant items and label items with fine-grained tags, and a user-item knowledge graph is built to discover the potential relationship between items and users. Finally, a differentiated recommendation strategy is put forward to make different recommendations for users with different characteristics. The experimental results show that through data augmentation algorithm to improve data quality, factorization machine model produces higher recommendation accuracy, the constructed knowledge graph can alleviate the cold start problem for recommendation, and the differentiated recommendation strategy has achieved better recommendations for both active and inactive users.

INDEX TERMS Recommender systems, differentiated recommendation framework, data augmentation, knowledge graph, factorization machine, cold start problem.

I. INTRODUCTION

E-commerce recommender systems (RSs) are personalized recommendation tools to enhance the overall marketing performance of e-commerce platforms. By establishing user-centric personalized marketing strategy, e-commerce RSs can help platforms provide the most needed information to the users at the most appropriate time, provide a more comfortable shopping experience, and enhance customer loyalty. The accuracy of the recommendation results is a key factor of determining the success or failure of the RSs. If the item recommended by the RSs is not needed by the user, the user will lose confidence in the RSs and regard the recommended information as spam. Therefore, it is crucial to improve the accuracy of RSs.

The ideal RSs will vary from person to person, i.e., provide a personalized recommendation based on the characteristics of each user. However, in practical applications, there are often many restrictions, so that the accuracy and breadth of the recommendation cannot reach an ideal state. Some classic issues such as sparsity problem and cold start problem are still main reasons hindering the development of RSs. In this paper, we will try to improve the accuracy and alleviate the cold start problem, and explore a new framework to deal with both issues at the same time.

For the issue of how to improve the recommendation accuracy, there are usually two ways. First, given a specified dataset, we can choose state-of-the-art models based on the characteristics of the dataset, or build a model suitable for

The associate editor coordinating the review of this manuscript and approving it for publication was Fabrizio Messina.

the dataset. Second, given a specified recommendation model, we can choose to improve the quality of the dataset. Based on our existing research work [1], factorization machine (FM) [2] model is qualified for this recommendation task. Therefore, we will not discuss how to choose a model in this paper, but focus on the second way. Actually, improving data quality is a relatively difficult problem because the features of the data itself cannot be lost and new hidden features need to be discovered. Research in this area belongs to the field of data augmentation. The goal of data augmentation is to provide higher quality input for subsequent tasks in order to get better output. There are two ways to augment data onto machine learning. One is to increase the size of the dataset. It is mainly used in situations where the dataset size is not sufficient to maintain model training. The other is to adjust the data without changing the size, i.e., enriching information, such as providing more labels or attributes. Different datasets require different data augmentation strategies. There is no unified way to adapt to all problems. We will focus on fashion dataset in this paper. In summary, the general principle of improving the recommendation accuracy are that making full use of the characteristics of the dataset and making full use of the existing methods, we will also deal with the issue in these two ways, i.e., augment data, and then choose an appropriate existing model.

For the issue of how to alleviate the cold start problem, the common idea is to add auxiliary information to the recommendation algorithm. Auxiliary information can enrich the description of users and items, enhance the mining ability of recommendation algorithms, and effectively compensate for the sparse or missing interaction information. With the development of Internet and mobile technologies, the types of auxiliary information are becoming more and more abundant, such as social networks, geographic information, and multimedia information. In this paper, we use knowledge graph as auxiliary information [3]. Knowledge graph has more excellent characteristics such as accuracy, diversity, and interpretability than other auxiliary information. We build a knowledge graph based on the characteristics of the fashion dataset to describe the relationship between items, items and users, and use a graph search algorithm to deal with the cold start problem.

By analyzing the behavior of users in e-commerce datasets, we find that in order to improve the recommendation accuracy and alleviate the cold-start problem, it is necessary to differentiate users and provide differentiated recommendation services. This will not only provide users with a better shopping experience, increase user stickiness, but also enhance platform competitiveness and form a benign shopping cycle. The contributions of this paper can be summarized as follows.

1) According to the characteristics of the fashion datasets provided by the current popular e-commerce websites, combined with the existing achievements in the field of deep learning, a data augmentation algorithm based on transfer learning is proposed. Then we apply factorization machine model on the processed data, and verify the validity of the data augmentation idea by recommendation accuracy.
2) The cold start problem is almost a problem for all recommendation models. We propose the idea of using knowledge graph to alleviate the cold start problem, and build a user-item knowledge graph on the fashion dataset to explore the implicit relationship between users and items.
3) A differentiated recommendation framework is proposed. In the framework, users are divided into active and inactive ones according to the frequency of users' activities. Different recommendation paths are designed for different user types to improve recommendation quality in terms of accuracy and cold start.

The rest of the paper is organized as follows. In Section 2 we review the related work. In Section 3, we show the overview of the proposed framework and briefly explain several important components of the framework. In Section 4, we propose a data augmentation algorithm based on transfer learning method, build a knowledge graph of user and item, and put forward a differentiated recommendation strategy. Section 5 shows the experimental results on amazon dataset. With the support of the proposed data augmentation algorithm, FM model obtains better prediction results of user ratings on items. Finally, we conclude the paper and give the future research directions in Section 6.

II. RELATED WORK

A. FASHION RECOMMENDATION

Fashion is a multi-billion-dollar industry. With the extensive research and application of artificial intelligence in various industries, it is a popular topic to find suitable intelligent applications for the fashion field. Among them, fashion recommendation is a hot research direction. RSs have been studied for a long time, but there are still many issues needing to be explored when RSs meet fashion.

The easiest way to think of is to apply the proven recommendation algorithm to the fashion filed. Hwangbo et al proposed a system called K-RecSys, which extends the typical project-based collaborative filtering algorithm. In order to better to reflect online and offline customer preferences, K-RecSys combines online product click data with offline product sales data to support decisions [4]. However, there are deficiencies in the application of the actual projects because the traditional recommendation method extracts features based on information such as attributes and feedback of users and items, and most of these features are based on text and numerical values. In the field of vision-based e-commerce, especially in the fast fashion field (such as Zara, H&M, Uniqlo, etc.), inquiries and recommendations of items on the online platforms are highly dependent on users' visual experiences [5]–[8].

As deep learning has been successfully applied in the fields of image detection, image segmentation, image annotation, and image generation [9], [10], it is also extended to RSs to break through the bottleneck of visual features in traditional fashion recommendation methods [9], [10]. The related research shows that visual features based on deep learning can bring higher accuracy RSs, and it is a trend to use visual features for fashion recommendation [6], [8], [11]. By considering the rich information of product images, Zhou et al. proposed a new fashion product recommendation method based on text and image mining technology [12]. To uncover trends in fashion, He et al. combined advanced visual features, user feedback, and changing trends within the community into One-Class Collaborative Filtering [8]. Tuinhof et al. developed a two-stage deep learning framework that combines traditional content-based recommendation to recommend similar styles of fashion images for users. In the framework, neural network classifiers are used as data-driven visual perception feature extractors [13]. Gu et al. presented a fashion coordinate system that considers user behavior and visual fashion style. It extends latent factor model (LFM) to handle user behavior characteristics and uses the denoising autoencoder network model to process visual features. By combining the two models, the recommendation accuracy is effectively improved and the cold start problem is solved [14].

Therefore, in the implementation of fashion recommendation in this paper, we also make full use of the existing research results of visual features and deep learning technology to provide support for better recommendations.

B. DATA AUGMENTATION

The quality of datasets can directly affect the performance of recommendation models. In order to improve the quality of datasets, data is usually cleaned and processed manually, but this is a very time-consuming task. How to use existing technologies to augment dataset is very significant. Data augmentation is mainly applied to two scenarios.

First, when a dataset is not sufficient to maintain normal training of a model, a larger dataset is generated by related techniques. The significance of data augmentation at this time is learning more with less. This method is mainly used in the field of deep learning. For example, when models such as AlexNet [15], GoogLeNet [16], and ResNet [17] are applied to image classification tasks, a large amount of data is required to train a better model. Generative adversarial networks (GANs) have been proved to be able to produce artificial data that are similar to the real data, and have been successfully applied to various image generation tasks as a useful tool for data augmentation [18], [19]. GANs is a commonly used data augmentation method in application scenarios. Shao et al. developed an auxiliary classifier GAN(ACGAN)-based framework to learn from mechanical sensor signals and generate realistic one-dimensional raw data [19]. Han et al. developed a conditional PGGAN-based data augmentation method of brain metastases detection using highly-rough annotation on MR images [15]. Tanaka et al. proposed to use GANs to generate artificial training data onto machine learning tasks [20]. Cui et al. used wasserstein GANs with a gradient penalty (WGAN-GP) to generate new samples based on existing SAR data, which can augment the sample number in training dataset [21].

Another scenario of data augmentation is that the dataset size is large enough, but the quality of the dataset is not good, i.e., labels are not inaccurate or there are insufficient attributes on the dataset. Dealing with this issue can be useful for subsequent tasks. However, such work in fashion recommendation field is still not taken seriously. We will focus on the data augmentation issue in this paper, make full use of the existing research results, and propose a universal data augmentation method to provide support for subsequent recommendation task.

C. RECOMMENDATION BASED ON KNOWLEDGE GRAPH

The knowledge graph was proposed by Google in 2012, and its original intention is to improve the user's search quality and experience. With the development and application of artificial intelligence technology, knowledge graph has gradually become one of the key technologies, and has been widely used in the fields of intelligent search, intelligent question and answer, personalized recommendation, content distribution, etc. Integrating knowledge graph as auxiliary information on RSs can effectively solve the sparseness and cold start problems of the traditional recommendation methods.

The application of knowledge graph in RSs covers almost many areas. In the beginning, it was mainly used in the field of film and music recommendation. For example, Oramas et al. built and utilized knowledge graphs to provide information to a hybrid recommendation engine based on a collection of documents describing music and sound projects. It uses tags and text descriptions to extract entities and links them to external graphics such as WordNet and DBpedia. These external graphics are then used to semantically enrich the initial data. These feature combinations are ultimately used to calculate the list of recommendation [22]. Catherine et al. developed a graph-based latent factor model called GraphLF, which combines the advantages of latent factorization and graphics and achieves significant performance gains on two large datasets, Yelp and MovieLens-100K [23]. Then there was the application of knowledge graph in the field of travel recommendation. Niaraki et al. combined the ontology with traditional recommendation algorithms to make recommendations for users [24]. The user profile is built according to the registration information to calculate the user interest and confidence, and the items are recommended according to the interaction behavior, combined with the degree of interest and confidence. Then the recommendation list is updated in real time by combining the ICEfaces framework in the GIS positioning system [24]. Later, the knowledge graph was also emerging in social networks. Karidi et al. constructed user interest lists by retrieving semantic information from tweets and used knowledge graph to calculate interest similarities between users in order to better provide personalized recommendations for users [25]. Deng et al. proposed a group recommendation method with knowledge graph

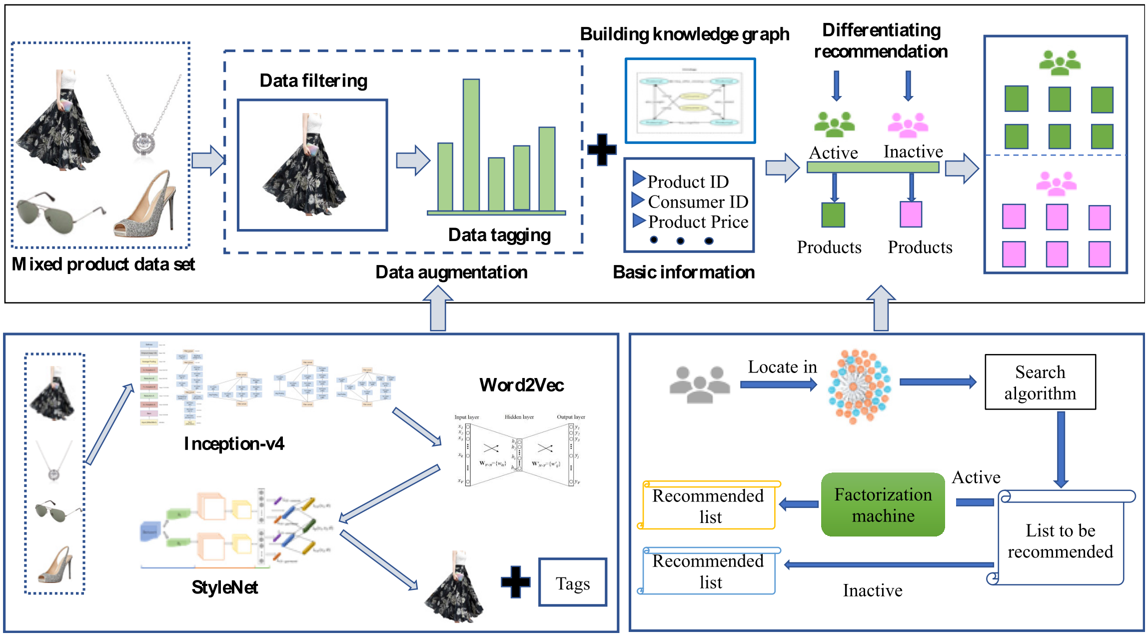

FIGURE 1. An illustration of our proposed framework. The figure consists of three parts. Top: Fashion recommendation. Bottom Left: Image feature extraction. There are two kinds of features, basic features and tags by data augmentation. Bottom Right: Differentiated recommendation. For active users, recommended list is produced by FM model, and for inactive users, recommended list is produced according to the knowledge graph.

enhancement, in which the knowledge graph is used to construct a comprehensive overview of the group and the information or knowledge that needs to be recommended [26]. In the academic field, Ayala-Gomez et al. proposed to use knowledge graph to extend the semantic features of a given abstract and combine them with other features such as indegree and recency to fit the learning ranking model, which is used to generate citation recommendation in academic search engines [27].

It can be seen from the above that the recommendation based on knowledge graph is quite mature, but its application in the e-commerce field is still in its infancy. This paper attempts to build a knowledge graph in a real e-commerce dataset, in order to provide a way to alleviate the cold start problem of recommendation tasks.

III. OVERVIEW OF THE FRAMEWORK

In order to recommend the most appropriate fashion items to users, it is necessary to understand the fashion items that a user is most interested in. A framework is needed to analyze the characteristics and classification of items and the relationship between users and items based on historical data, and then predict the possible purchases of users. Fig. 1 shows the proposed system diagram which includes four core components.

1) Data filtering. We observed that in the product dataset, fashion products include not only clothing items, but also other fashion items such as watches, shoes or glasses. If the function of the system is to recommend clothing for users, the unwanted products are best filtered out in advance. In view of the good effect of the deep learning method in the field of image classification, we will implement this process by applying the deep learning model and using the transfer learning method. If the system is used to recommend jewelry, products not related to jewelry will be filtered out.

2) Data tagging. We also observed that the dataset only includes coarse-grained classification information and attribute information, and there is no fine-grained classification information for each item. Classification information is proven to be very important for recommendation, which provides a feature for recommendation algorithms. This paper will augment data from the two aspects of data filtering and data tagging.

3) Building knowledge graph. The Web2.0 and web3.0 eras have led to an increased interaction between users and websites. Uncertain relations between users, items, and their relations can be transformed into a knowledge graph. The role of building a knowledge graph is to solve the cold start problem and increase the breadth of recommendations.

4) Differentiating recommendation. Based on the rating records, the user behaviors can be analyzed and then the users are divided into active and inactive ones. Combined with FM model and knowledge graph, a differentiated recommendation strategy is used to provide different services for different users so that the whole performance of the system can be improved.

IV. METHODOLOGIES

A. DATA AUGMENTATION

Accuracy is often one of the important criteria for measuring the quality of RSs. The quality of dataset directly affects the accuracy of RSs. In fashion field, there are not many datasets to choose from, and the most popular used dataset is Amazon

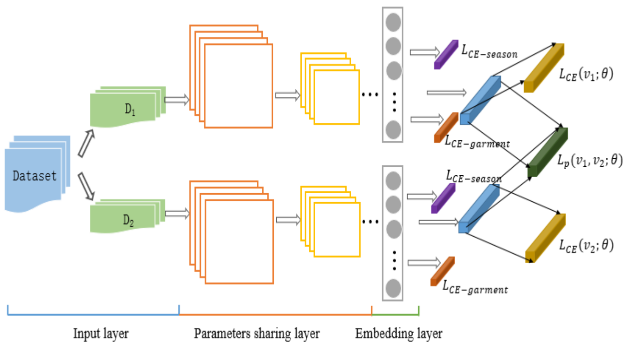

FIGURE 2. The structure of model StyleNet based on multi-task representation learning framework. The dataset is divided into two subsets to train the model. This approach does not increase the complexity of learning because both models share the same parameters.

fashion dataset. However, Amazon's data information is not rich enough. The following problems occur when performing recommendation tasks based on it. 1) The classification of items is too coarse to be accurately judged. 2) There are too few available attributes for items, which is not conducive to content-based or collaborative filtering algorithms. In response to these two problems, a data augmentation idea is put forward. We implement it in two aspects, coarse granularity and fine granularity.

Firstly, data filtering. This paper focuses on fashion clothing recommendation in the Amazon dataset, so how to identify fashion clothing from large-scale item images is the key. If fashion jewelry recommendation is studied, the method is the same. Google proposed Inception-v4 networks [28] which achieves 3.08% top-5 error on the test set of the ImageNet classification (CLS) challenges. Here we use the trained model directly to process each image of the dataset, and then select the top-3 tags as the label for each image. Next, these three tags are used as input to the well-known Word2Vec model on the field of natural language processing to produce an output vector [29]. The output vector is compared to the predefined vectors for the classification. When the vector with the smallest distance is found, the classification to which the vector belongs is the final label corresponding to the output vector. If the label is clothing, the item will be selected, otherwise it is filtered out. The predefined vectors include 'clothes', 'shoes', 'jewelry', 'ornaments', 'clock', and 'outdoor'.

Secondly, data tagging. After the required fashion items are selected from the dataset, they will be labeled with fine-grained tags. These tags will support subsequent recommendation algorithms. StyleNet is a style representation learning model based on the deep neural network, which is proposed in our previous work [30]. It is a multi-tasking clothing classification framework that can divide items by season, type, and style. StyleNet has been trained in our previous work and is taken directly here for application.

Fig. 2 shows the structure of StyleNet, which consists of four parts, input layer, parameters sharing layer, embedding layer and constraint layer. The objective function L_{all} is expressed as follow:

$$L_{all} = \lambda_1 L_{season} + \lambda_2 L_{p-style} + \lambda_3 L_{garment} \tag{1}$$

where λ_1, λ_2, λ_3 are weights, L_{season} and L_{garment} represent the corresponding cross entropy losses, and $L_{p-style}$ is the confusion loss with distance constraint [30].

Algorithm 1 shows the process of data augmentation. The input is the original dataset. The output is the dataset containing fine-grained tags and it belongs to a certain category.

Algorithm 1 Data Augmentation Algorithm

```
Input:rawDS        //Raw dataset with unknown category
Output:augDS    // Clothing items with additional tags
augDS ← {}
for each item in rawDS do
get image_vector from item.image by CNN
input image_vector to Inception-v4
top3_item_tags←get top3 classification tags
from Inception-v4
distance←{}
for each word in predefinedWordList do
        get word's word_vector by Word2Vec
        distance_tmp←0
        for each tag in top3_item_tag do
            get tag's tag_vector by Word2Vec
            distance_tag←calculate distance
  between tag_vector and word_vector
            distance_tmp←distance_tmp
 +distance_tag
            distance[word]←distance_tmp
    end
end
if min(distance.values).key is equal to
     "clothes" do
       input image_vector to StyleNet
       tags_StyleNet←get tags from StyleNet
       end
augDS←(item, tags_StyleNet)
end
return augDS
```

The time complexity of Algorithm 1 is analyzed below. Assuming the number of items in dataset *rawDS* is t, then the outermost loop count is t. In the second layer of loops, the length of the predefined word is 6. In the third layer of loops, the length of *top3_item_tag* is 3. Since $6 << t$ and $3 << t$, the time complexity of the whole algorithm is O(t).

However, in terms of time consumption, because it is using up to 3 models, the time consumption of every cycle is longer than we think. Because of this, in the actual experiment, we split the algorithm into three parts so that we can implement them by three threads to process in parallel. The first part is connected to the Inception-v4 model to

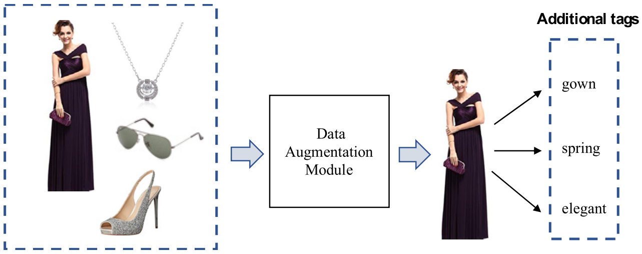

FIGURE 3. An instance of data augmentation. After data augmentation module, we get clothing images and the tags related to them. These tags are learnt by StyleNet. Tag gown is used to describe the type, tag spring describes the season, and tag elegant describes the style.

get *top3_item_tag*, the second part is connected to Word2Vec to filter out the clothing image, and the third part is connected to the StyleNet model to get the additional tags. In this case, the experimental process becomes more intuitive and controllable.

Finally, to better demonstrate the effectiveness of data augmentation algorithm, we give an example in Fig. 3.

B. RECOMMENDATION WITH FM

Since FM model can effectively solve the sparsity problem of high-dimensional data features combination of high prediction accuracy and computational efficiency, it has been widely studied and applied to the field of click-through-rate (CTR) prediction and RSs. The second-order FM is defined as follow:

$$\hat{y}(x) = w_0 + \sum\nolimits_{i=1}^{n} w_i x_i + \sum\nolimits_{j=1}^{n} \langle v_i, v_j \rangle x_i x_j \tag{2}$$

where n is the number of instances, k is the dimensionality of the factorization and $\Theta = \{\omega_0, \omega_{\mathrm{i}}, \omega_{\mathrm{ij}}\}$ are the model parameters.

For deriving a point estimator of the model parameters, the task is to minimize an objective consisting of loss function l and regularization as follow:

$$OPT\ (S) = \arg\min_{\Theta} \left(\sum_{(c,x,y)\in S} cl(\hat{y}\,(x|\Theta)\,,\, y) + \frac{1}{2} \sum_{\theta\in\Theta} \lambda_\theta ||\theta||^2 \right) \tag{3}$$

where the loss function l can be the least-squares for regression or logistic loss for binary classification.

FM can be trained and predicted with a complexity O($k*n$), making it a quite efficient model. There are two reasons for using FM model in this paper. 1) In order to improve the accuracy of the recommendation, we need to choose an excellent model, and FM model has been proven in our previous work. 2) In order to verify that the data augmentation algorithm in 4.1 is valid, we need to use a recommendation model. After the data augmentation operation, the quality of data has been improved, and it is necessary to verify whether the positive feedback is generated for the recommendation.

Here, FM is used for binary classification task. If the score in the original dataset is 1, 2, or 3, the classification is set to -1, which means that the user is not interested in the item.

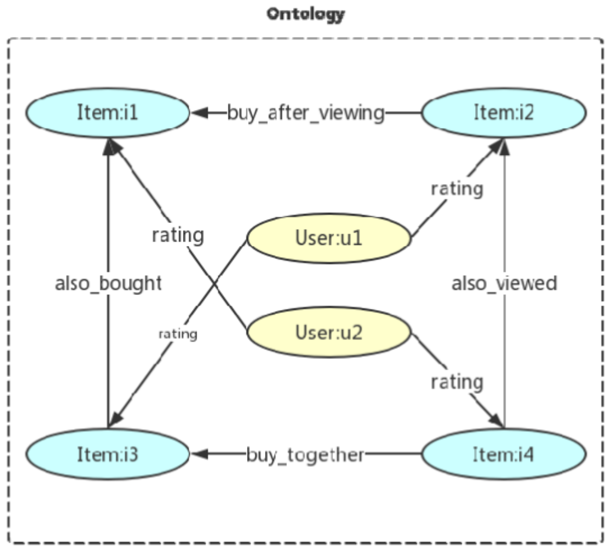

FIGURE 4. Ontology definition. There are four kinds of relations between items, also_bought, buy_together, buy_after_viewing, and also viewed. The relation between user and item is rating score.

If the score in the original dataset is 4 or 5, the classification is set to 1, which means that the user is interested in the item.

In the verification process, we did a set of comparative experiments, i.e., data augmented and original data were separately processed by FM. Before data is processed, the input of FM includes user id, item id, and gender. After data is processed, the input includes user id, item id, item tags, season, style, and gender. If the output is -1, then it will not be recommended. If the output is 1, it is recommended. The comparison of the accuracy measurement can be used to illustrate whether the idea of 4.1 is feasible.

C. KNOWLEDGE GRAPH BUILDING ON DATASET

The knowledge graph is a structured semantic knowledge base. It has been widely used in many fields, such as Google search, Baidu search and LinkedIn economic graph. We use knowledge graph in this paper to alleviate the cold start problem. Although the original dataset is previously augmented in the framework, there are no historical data that can be analyzed and referenced for some users who have no or few traces on the site. After trying and considering, we finally chose to build a user-item knowledge graph.

Building a knowledge graph from the bottom up is a huge project involving information extraction, knowledge fusion, knowledge processing, and knowledge storage. Given that the data in the dataset are semi-structured, the process of building a knowledge graph is relatively straightforward. It mainly includes the following three steps.

The first is to define the ontology. For a simple domain, it is simpler and more efficient to build from the bottom up. This paper is based on this idea of construction, so the definition of ontology has become the first step.

As shown in Fig. 4, the knowledge graph we built contains two concepts. One concept is Item and the other is User. There is only one interaction between User and

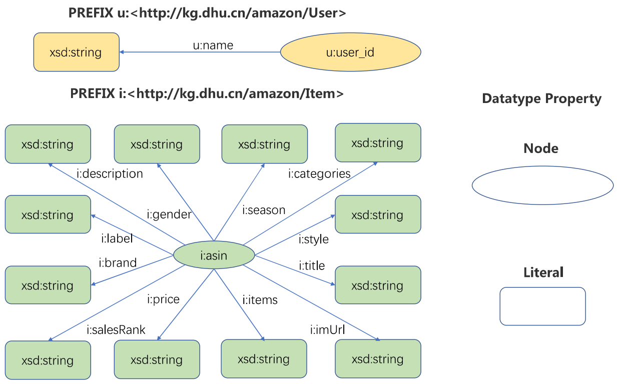

FIGURE 5. **Concepts of User and Item. The oval represents node and the rounded quadrilateral represents literal. There are two kinds of nodes. The yellow oval represents the User, and green one represents the Item.**

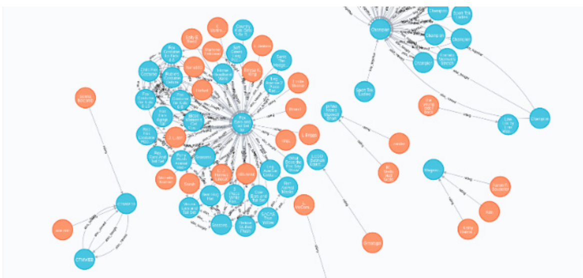

FIGURE 6. **Instance of user-item knowledge graph. The red nodes represent the users, the blue nodes represent the items, and they interact through our predefined relationships.**

Item, rating. There are four interactions between items and items. They are buy_after_viewing, also_viewed, buy_together, and also_bought.

The second step is to define concepts. Fig. 5 shows the definition of two concepts, User and Item.

The third step is to extract information, including entity extraction and relationship extraction. After the information is extracted, the knowledge graph can be built. The built knowledge graph is stored in the current mainstream graph database neo4j. The resulting knowledge graph contains 1,001,229 nodes and 3,359,804 relationships. Among them, there are 283,385 Item nodes, 717,914 User nodes, 980,753 rating relationships, 952,403 also_viewed relationships, 1,358,655 also_bought relationships, 67,928 buy_together relationships, and 55 buy_after_viewing relationships. The final knowledge graph is shown in Fig. 6.

D. DIFFERENTIATED RECOMMENDATION STRATEGY

By analyzing the dataset, it was found that only a small number of users had a high number of rating records, and most users had very few rating records. After data filtering, the statistical information of Amazon fashion clothing dataset is shown in Table 1. In fact, almost all large e-commerce websites can provide such information.

It can be seen from the statistics in Table 1 that the number of users with less than 5 rating records is 702,558, accounting for up to 97.78%, and the number of users with rating records between 5 and 10 is 13,766, accounting for 1.92%, and the number of users with rating records higher than 10 is 2,162, accounting for only 0.30%.

TABLE 1. **Statistic information of Amazon fashion dataset after data augmentation operation.**

User type	Number of users	Number of rating records	Number of rating record per user
all	718,486	983,303	1.37
Rating records<5	702,558	849,050	1.21
5≤Rating records<10	13,766	102,207	7.42
Rating records≥10	2,162	32,046	15.07

Definition 1 (Active Users and Inactive Users): In terms of the general cold start problem, a user with a small number of rating records can be considered as a new user or an inactive user. Here, users with less than 10 rating records are treated as inactive users. Because such users have very few rating records, many recommendation algorithms are not ideal for them. One practical solution is to add some auxiliary information. The auxiliary information used in this paper is the knowledge graph. A user whose number of rating records is greater than 10 is called an active user. For this type of users, we will not only combine the knowledge graph, but also use FM model to enhance the recommendation experience of them.

A differentiated recommendation strategy for different user groups is proposed in this paper. For active users, FM model and knowledge graph are used to provide high-quality recommendation services, and for inactive users, only knowledge graph is used for simple recommendation. The differentiated recommendation strategy algorithm is shown in Algorithm 2.

Assume that the number of users is u, the maximum number of items purchased by a user is u_1, and the number of active users is u_2. In algorithm 2, the outermost loop count is u. There are two loops in the second layer. One is the traversal of the user purchases and the loop count is u_1. The other is the traversal for active users and the loop count is u_2. Obviously, $u_1 << u$ and $u_2 << u$, so both u_1 and u_2 can be regarded as constants. Then we can conclude that the time complexity of Algorithm 2 is O(u). It is worth noting that during the traversal of user purchases, Algorithm 2 performs a tree-based depth-first query, assuming its depth is d. According to the rules of the depth-first traversal algorithm, the complexity is O(d^2). The performance of deep search is the key to this algorithm. The deeper the depth, the more time-consuming the search. Considering the time factor of the experiment, we set 2 for the search depth.

V. EXPERIMENTS AND EVALUATION

A. DESCRIPTION OF THE DATASET

There are not many e-commerce datasets available at present. The dataset used in this paper is Amazon dataset, a classic e-commerce dataset. It includes 142.8 million reviews (ratings, text, helpfulness votes) spanning from May 1996

Algorithm 2 Differentiated Recommendation Algorithm

Input:*user, d*//d is the search depth
Output*: recLists // a set of items*

```
locate user in the predefined KG
items_list1 ←get the corresponding purchased items
items_list2 ←{ }
for each item in items_list1 do
    tmp ←perform a depth search with depth d
    items_list2 ←items_list2+ tmp
    sort and combine items_list2
end
recLists ←{ }
if user is an active user do
    for each item in items_list2 do
        let item access the Pre-trained FM and get classification
        if classification is equals to 1 do
            recLists ←items_list2+item
        end
    end
else
    recLists ← items_list2
end
return recLists
```

to July 2014, item metadata (descriptions, category information, price, brand, and image features), and links (also viewed/also bought graphs). In the experiment, we chose some subsets from the raw dataset. In 'metadata' subset, it includes descriptions, price, sales-rank, brand info, and co-purchasing links. We use it to augment data and build knowledge graph. We also use subset of 'Ratings only'. These are not metadata or reviews and it only includes some tuples (user, item, rating, timestamp). This subset is used to verify the feasibility of data augmentation idea and used by FM model for recommendation.

B. EXPERIMENT SETTINGS

The hardware platform running experiments is Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz, Nvidia®Tesla K80 GPU, 128GB RAM, 1TB hard drive, 64-bit Ubuntu 16.04 operating system. The algorithms are implemented in python language and they run under Caffe platform. The following will be divided into three parts to conduct experiment and judge the proposed algorithms and framework.

C. IMPACT OF DATA AUGMENTATION ON RECOMMENDATION ACCURACY

First, we test the effect of the data filtering operation. The total number of items in Amazon dataset is 1,503,384. The number of clothing items obtained after data filtering operation is 283,385. From the difference in numbers, we can see that such filtering operation cannot be done manually, but only by means of computers. The test data was obtained using random sampling method. A detailed description of the method is as follows. We randomly sampled 100 items from the processed dataset, manually judged whether they were clothing products, and then calculated the probability of clothing items. The operation was repeated 10 times and the average probability is 97.2%. According to this result, we can see that the data filtering idea proposed in this paper is feasible and the effect is ideal. The high accuracy of data filtering provides guarantee for the subsequent tagging of StyleNet model.

TABLE 2. Accuracy of prediction.

Model	Random model	MF without tagging	FM without tagging	FM without tagging
DS1	51.62%	75.99%	77.33%	80.43%
DS2	52.46%	76.26%	76.42%	79.67%
DS3	51.87%	76.20%	77.12%	79.65%
DS4	50.64%	75.38%	76.54%	81.25%
DS5	52.21%	76.43%	77.24%	80.02%

Next we verify the impact of data augmentation on the recommendation. We conduct two sets of comparative experiments. One group is an experiment without data augmentation. This group of experiments runs a total of three models. This group of experiments is used to compare and explore the effects of different models on the original data. Another set of experiments is to verify whether the data augmentation algorithm has a positive impact on the recommendation, so they are conducted after data augmentation. The comparison model here selected is FM because of its sensitive properties for attribute interaction. Our processing of the data is to use experimental data for processing binary classification tasks, where records with scores of 1-3 are considered negative samples, and records with scores of 4-5 are considered positive samples. Among them, the hidden vector size of the FM model is set to 33, the number of iterations is 80, and the random gradient descent (SGD) algorithm is used for training. The experiment is carried out five times. Each training set and test set were extracted by a random method. The ratio of the training set to the test set was 8:2. The final experimental results are shown in Table 2.

From Table 2 we can draw the following conclusions. First, compared with the random model, the accuracy of FM model is significantly improved, which proves that FM is an excellent recommendation model. Secondly, the accuracy of FM and matrix factorization (MF) is basically the same without adding tags. This is expected and interpretable, because the improvement of FM model is mainly reflected in the interaction between attributes, and if there is no attribute interaction, the two are basically at the same level. Finally, after our data augmentation process, the products have been tagged with new tags. At this time, the property interaction characteristics of FM can be used. From the experimental results, it can be seen that the proposed method brings 3.27% improvement.

TABLE 3. Hit rate, coverage and time consuming of KG.

Depth	Hit rate	Coverage	Time(s)
1	16.67%	62	0.05
2	43%	2547	0.57
3	100%	112044	27.89
4	100%	5130298	1512.24
5	100%	>5130298	---

Although the improvement is not very significant, it is enough to prove that our data augmentation ideas are feasible and have certain significance for improving the recommendation accuracy.

D. IMPACT OF KNOWLEDGE GRAPH ON COLD STARTS PROBLEM

The knowledge graph can be used to solve the cold start problem of RSs and increase the breadth of recommendations. We adopt three dimensions to evaluate the impact of knowledge graph on cold starts problem, they are hit rate, coverage, and time consuming. For hit rate, our test method here is to select 80% of the products purchased by users, search for the remaining 20% of the products through the knowledge graph, find the intersection of the predicted result and 20% of the products as the number of hits, and finally calculate the hit rate, that is, the number of hits/20% of the items. And the coverage examines the number of items covered by its search. This indicator overlaps with the hit rate but is independent in order to more intuitively display the breadth of the search. Time consuming is the time it takes to search. The detailed experimental steps are as follows. Each time 5 users are randomly selected, each user's purchase item is obtained, and then searches for depths of 1, 2, 3, 4, and 5 are performed separately for these items. Finally, to calculate the hit rate, coverage and time consumption for each search, the experiment was performed a total of 5 times and the mean was taken. The experimental results are shown in Table 3.

As can be seen from Table 3, as the depth of search increases, the hit rate, coverage and time consumption increase, and the coverage and time consumption increase exponentially. Among them, the hit rate and time consumption are our main concerns, and these two indicators have a great impact on the user experience. It can be seen from the table that when the depth is 3, the search time has become unacceptable. Therefore, based on the time considerations and the certain hit rate, we finally choose 2 as the search depth.

E. RESULTS OF DIFFERENTIATED RECOMMENDATION

According to the above experimental results, the data augmentation algorithm proposed in this paper is effective, and the application of knowledge graph is also feasible. For the proposed differentiation strategy, the evaluation strategy adopted here is an illustrative method. We randomly select some users from active users and inactive user groups, and provide differentiated recommendation strategies for them.

In Amazon dataset, there is an inactive user A3ETDO8R97 KK39 with only three purchase records. By studying the corresponding imUrls, it can be inferred that this user's shopping preference is male clothing. Then we performed a depth search of depth 2 based on the established user-item knowledge graph. The operation takes 0.004s and the returned number is 1009. The recommendation strategy is to recommend 10 items to a user each time. The four relationships of buy_together, also_bought, also_viewed and buy_after_viewing were randomly combined according to a certian weight ratio.

There is an active user A1317ENKFASY9W with 23 purchase records. By studying the corresponding imUrls, it can be inferred that this user's shopping preference is female clothing. Then we performed a depth search of depth 2 based on the established user-item knowledge graph. The operation takes 1.7068s and the returned number is 9836. The item data of 9836 is input to the previously trained FM model, and the number of recommended products returned is 2952. Finally, the recommendation strategy is to recommend 10 items to a user each time according to the four relationships of buy_together, also_bought, also_viewed and buy_after_viewing.

VI. CONCLUSION

High-quality recommendation service has become one of the strongest competitions of e-commerce. With the development of technology, the ways taken to solve the problems such as data sparseness and cold start are becoming more and more diverse. In this paper, we put forward a data augmentation idea by integrating the existing research results of deep learning, combined with factorization machine model to provide high-quality data support for improving recommendation accuracy. Although the experiment was based on Amazon datasets, this idea can be migrated to similar fashion datasets. Then we use the heterogeneous network built by knowledge graph and its interpretability features to solve the cold start problem of recommendation. Finally, through the analysis of the characteristics of different user groups, from the perspective of diversity service, a differentiated recommendation strategy is proposed.

By comparing the recommendation effects before and after the data augmentation processing in the experiment, the importance of data quality is more obvious. StyleNet is a representation learning model that is primarily designed for classification tasks in our previous work. Since wrong data tags will directly affect the subsequent recommendation task, we will further adjust StyleNet model to obtain more fine-grained and accurate tags in the future work. When integrating knowledge graph into the recommendation strategy, we only use the general graph algorithm, and do not go deep into the connection between entities. In the future, ideas such as meta-path [31] will be considered to further explore the benefits of knowledge graph.

REFERENCES

[1] C. Yan, Q. Zhang, X. Zhao, and Y. Huang "An intelligent field-aware factorization machine model," in *Proc. Int. Conf. Database Syst. Adv. Appl.*, New York, NY, USA, Mar. 2017, pp. 309–323.

[2] S. Rendle, " Factorization machines with libFM," *ACM Trans. Intell. Syst. Technol.*, vol. 3, no. 3, May 2012, Art. no. 57.

[3] R. Verborgh, M. V. Sande, O. Hartig, J. Van Herwegen, L. De Vocht, B. De Meester, G. Haesendonck, and P. Colpaert, "Triple pattern fragments: A low-cost knowledge graph interface for the Web," *J. Web Semantics*, vols. 37–38, pp. 184–206, Mar. 2016.

[4] H. Hwangbo, Y. S. Kim, and K. J. Cha, "Recommendation system development for fashion retail e-commerce," *Electron. Commerce Res. Appl.*, vol. 28, no. 3, pp. 84–101, Mar./Apr. 2016.

[5] J. Mcauley, C. Targett, O. Shi, and A. van den Hengel, "Image-based recommendations on styles and substitutes," in *Proc. 38th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, New York, NY, USA, Aug. 2015, pp. 43–52.

[6] C. Bracher, S. Heinz, and R. Vollgraf, "Fashion DNA: Merging content and sales data for recommendation and article mapping," in *Proc. 22nd ACM SIGKDD Workshop 'Mach. Learn. Meets Fashion'*, New York, NY, USA, 2016, pp. 1–10.

[7] R. He, C. Lin, J. Wang, and J. McAuley, "Sherlock: Sparse hierarchical embeddings for visually-aware one-class collaborative filtering," in *Proc. 25th Int. Joint Conf. Artif. Intell. (IJCAI)*, New York, NY, USA, 2016., pp. 3740–3746.

[8] R. He and J. Mcauley, "Ups and Downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering," in *Proc. 25th Int. Conf. World Wide Web (WWW)*, Piscataway, NJ, USA, Apr. 2016, pp. 507–517

[9] J. Donahue, M. Khapra, and V. C Raykar, "Decaf: A deep convolutional activation feature for generic visual recognition," in *Proc. 31st Int. Conf. Mach. Learn. (ICML)*, Piscataway, NJ, USA, Jan. 2014, pp. 988–996.

[10] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New York, NY, USA, Jun. 2014, pp. 512–519.

[11] Z. Lin, "An empirical investigation of user and system recommendations in e-commerce," *Decision Support Syst.*, vol. 68, no. 12, pp. 111–124, Dec. 2014.

[12] W. Zhou, P. Y. Mok, Y. Zhou, Y. Zhou, J. Shen, Q. Qu, and K. P. Chau, "Fashion recommendations through cross-media information retrieval," *J. Vis. Commun. Image Represent.*, vol. 61, no. 5, pp. 112–120, May 2019.

[13] H. Tuinhof, C. Pirker, and M. Haltmeier, "Image-based fashion product recommendation with deep learning," in *Proc. Int. Conf. Mach. Learn., Optim., Data Sci. (LOD)*, New York, NY, USA, Feb. 2018, pp. 472–481.

[14] S. Gu, X. Liu, L. Cai, and J. Shen, "Fashion coordinates recommendation based on user behavior and visual clothing style," in *Proc. 3rd Int. Conf. Commun. Inf. Process.*, New York, NY, USA, Nov. 2017, pp. 185–189.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "The elements of fashion style," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, New York, NY, USA, 2012, pp. 1097–1105.

[16] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New York, NY, USA, Jun. 2015, pp. 1–9.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New York, NY, USA, Jun. 2016, pp. 770–778.

[18] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, New York, NY, USA, 2014, pp. 2672–2680.

[19] S. Shao, P. Wang, and R. Yan, "Generative adversarial networks for data augmentation in machine fault diagnosis," *Comput. Ind.*, vol. 106, no. 4, pp. 85–93, Apr. 2019.

[20] S. Milz, T. Rudiger, and S. Sus, "Aerial GANeration: Towards realistic data augmentation using conditional GANs," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Berlin, Germany, Sep. 2018, pp. 59–72.

[21] Z. Cui, M. Zhang, Z. Cao, and C. Cao, "Image data augmentation for SAR sensor via generative adversarial nets," *IEEE Access*, vol. 7, pp. 42255–42268, 2019.

[22] S. Oramas, V. C. Ostuni, T. D. Noia, X. Serra, and E. D. Sciascio, "Sound and music recommendation with knowledge graphs," *Trans. Intell. Syst. Technol.*, vol. 8, no. 2, Jan. 2017, Art. no. 21.

[23] R. Catherine and W. Cohen, "Personalized recommendations using knowledge graphs: A probabilistic logic programming approach," in *Proc. 10th ACM Conf. Recommender Syst.*, New York, NY, USA, Sep. 2016, pp. 325–332.

[24] A. S. Niaraki and K. Kim, "Ontology based personalized route planning system using a multi-criteria decision making approach," *Expert Syst. Appl.*, vol. 36, no. 2, pp. 2250–2259, Mar. 2009.

[25] D. P. Karidi, Y. Stavrakas, and Y. Vassiliou, "Tweet and followee personalized recommendations based on knowledge graphs," *J. Ambient Humanized Comput.*, vol. 9, no. 6, pp. 2035–2049, Nov. 2018.

[26] W. Deng, P. Zhu, and J. Ma, "Enhancing group recommendation by knowledge graph," in *Proc. 22nd Pacific Asia Conf. Inf. Syst. (PACIS)*, Providence, RI, USA, 2012, pp. 3570–3577.

[27] F. Ayala-Gómez, B. Daróczy, A. Benczúr, M. Mathioudakis, and A. Gionis, "Global citation recommendation using knowledge graphs," *J. of Intell. Fuzzy Syst.*, vol. 34, no. 5, pp. 3089–3100, May 2019.

[28] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proc. 31st AAAI Conf. Artif. Intell.*, Menlo Park, CA, USA, Feb. 2017, pp. 4278–4284.

[29] K. W. Church, "Word2Vec," *Natural Lang. Eng.*, vol. 23, no. 1, pp. 155–162, 2017.

[30] C. Yan, L. Zhou, and Y. Wan, "A multi-task learning model for better representation of clothing images," *IEEE Access*, vol. 7, pp. 34499–34507, 2019.

[31] G. Fu, Y. Ding, A. Seal, B. Chen, Y. Sun, and E. Bolton, "Predicting drug target interactions using meta-path-based semantic network analysis," *BMC Bioinf.*, vol. 17, no. 1, p. 160, Dec. 2016.

CAIRONG YAN received the Ph.D. degree in computer science from Xi'an Jiaotong University of China, in 2006. She is an Associate Professor with the School of Computer Science and Technology, Donghua University of China. Her research interests include cloud computing, big data, and machine learning.

YIZHOU CHEN is currently pursuing the master's degree with the School of Computer Science and Technology, Donghua University of China. His research interests include machine learning and recommender systems.

LINGJIE ZHOU is currently pursuing the master's degree with the School of Computer Science and Technology, Donghua University of China. His research interests include machine learning and image processing.