Received September 26, 2018, accepted October 3, 2018, date of publication October 9, 2018, date of current version October 31, 2018.

*Digital Object Identifier 10.1109/ACCESS.2018.2875026*

# Combining Sentiment Analysis With a Fuzzy Kano Model for Product Aspect Preference Recommendation

**JINMING ZHANG[1], DEXIN CHEN[2], AND MIN LU[1]**

[1]School of Information and Security Engineering, Zhongnan University of Economics and Law, Wuhan 430073, China
[2]Faculty of Education, Hubei University, Wuhan 430073, China

Corresponding author: Min Lu (lumingok@163.com)

This work was supported by the Hubei Provincial Natural Science Foundation of China under Grant 2018CFB315.

**ABSTRACT** Item-based collaborative filtering (ItemCF) is a proven algorithm in recommendation systems that is based on a neighborhood algorithm, but it neglects the influence of sentiment in different aspects. However, customers always express their opinion in reviews, and these personalized data will influence the recommendation effect. This paper proposes an aspect sentiment collaborative filtering algorithm (ASCF), which combines sentiment analysis with a fuzzy Kano model. ASCF obtains the users' different attitudes toward aspects of the product by fine-grained sentiment analysis from the user's purchase records, and then analyzes the user's degree of desire and importance for each feature based on the fuzzy Kano model, proposing a novel similarity measure method with user preferences for a collaborative filtering algorithm. Experiments with Amazon data sets show that ASCF effectively improves the precision of ItemCF and opinion-enhanced collaborative filtering; it provides higher recommendation precision and fewer product recommendations at the similarity precision. The experiments use the smartphone catalog as an example to analyze the aspect-characteristic words distribution matrix.

**INDEX TERMS** Collaborative filtering, opinion mining, sentiment analysis, Kano model.

## I. INTRODUCTION

Currently, facing an abundance of data, it is extremely difficult to find the information users truly need. When no longer facing the problem of information deficiency, overload becomes the new issue, so it is especially significant to establish an effective filtering mechanism to find useful information. As a result, researchers increasingly focus on researching recommendation systems. Collaborative filtering is a classic filtering algorithm for recommendation systems that can be divided into two categories: user-based collaborative filtering (UserCF) [1] and item-based collaborative filtering (ItemCF) [2], [3]. UserCF makes recommendations by finding users similar to the target user and uses these similar users' favorite items to make recommendations, while ItemCF tries to find products similar to the items that the user has already purchased according to their purchasing and browsing record. Although ItemCF has been researched by researchers from the perspective of product features, it is only limited to the dimensionality of the similarity of item features. Few people consider the relationship between item features and user requirements, which leads to the similarity not being sufficiently accurate to affect the recommended precision.

Playing an increasingly important role in people's daily lives, e-commerce makes it easier for us to obtain the subjective opinion data of users, such as online reviews. For the utilization of ratings as a measure of similarity for recommendations, the ratings only reflect the user's overall attitude toward the product, rather than reflecting the user's personalized factors, so the effect of the recommendation still requires improvement. Nevertheless, because the comments are a reflection of the user's preference for the product, together with the positive or negative attitude of the product, methods based on content mining can provide more accurate recommendations than the score-based methods. Product reviews will have a more profound impact on the user's purchasing behavior.

Considerable work has been done on recognizing product features and sentiment in user reviews, though there are relatively few studies on recommendation systems considering users' requirements and preferences. Collaborative filtering algorithms essentially measure the similarity of items or users, so researchers hope to make progress in making people or items more similar. Nevertheless, researchers have ignored the user's preferences for various aspects of the product characteristics. In other words, users have different requirements and preferences for different aspects of the product, so a collaborative filtering algorithm ignores the influence of user preferences by only considering the similarity of items instead of the items' aspects. Although this issue does not have a tremendous impact on the comprehensive effectiveness of the recommendation system, it is undoubtedly a concern. Please consider such a case where the user does not want item B, which is similar to A, to be recommended. This user gives a high-level, comprehensive evaluation for item A but expresses dissatisfaction toward one or more features of A. When item B is similar to A and has these features, it means that the user may expect B not to be recommended. For example, if the user is satisfied with the appearance of the phone A, but feels the price is too high, then the similar cell phone B should not be recommended if it is equally expensive. For this reason, under the inspiration of the Kano model, this paper proposes a collaborative filtering algorithm, ASCF, that combines the affective factors. The algorithm calculates the user's polarity of product features through sentiment analysis. Combining fuzzy Kano, the features are grouped by degree of need, and the high-demand features are used to measure the product similarity instead of all the features to improve the precision of the recommendation system.

This paper introduces the content in the following sequence. The second part reviews the literature, the third part presents a collaborative filtering algorithm that combines requirements and sentiment, the fourth part demonstrates the experimental design, the fifth part presents the evaluation, and the last part discusses the limitations of the research and further work.

## II. GUIDELINES FOR MANUSCRIPT PREPARATION

### A. OPINION MINING

Sentiment analysis, also known as opinion mining or comment mining, refers to a method of finding consumer sentiment through techniques such as natural language processing, machine learning and semantic analysis [4], [5].

Users diffuse and obtain information through various channels such as Weibo and websites [6]. In e-commerce, user's comments can be analyzed to find the attitude distribution, thus providing more personalized services for users. Currently, there are many studies on sentiment polarity classification. According to the granularity, sentiment polarity classification can be divided into fine-grained and coarse-grained sentiment analysis.

Document-level and sentence-level sentiment analysis belong to coarse-grained sentiment analysis. Document-level sentiment analysis generally determines the overall sentiment polarity of the text, which expresses the overall attitude of the positive or negative aspects. However, online reviews contain different attitudes toward different product aspects; coarse-grained analysis ignores this distinction, which leads to imprecise recommendations. The general approach to the study of text at the document-level is to transform data into a bag-of-words model, which performs textual representations and classification predictions through feature extraction and statistical learning methods. The current mainstream approach to textual sentiment classification is machine learning for the text, which is represented through the vector space model (VSM) for sentiment classification. Therefore, feature extraction is the core issue of this method. Ng *et al.* [7] argued that adding bigram and trigram features to unigram features can improve the accuracy of sentiment classification. However, if only one of them is selected as the feature, the classification accuracy declines while the order increases. Some researchers have shown that adjectives, nouns and adverbs tend to be used in semantic expression, and part-of-speech tagging and can be used to extract sentiment characteristics [8]–[10].

In essence, the sentence can be regarded as a short document. Currently, there are supervised learning, unsupervised learning, semisupervised learning and dictionary-based approaches used in sentence-level sentiment analysis. Compared with the document-level sentiment analysis, the sentence-level sentiment analysis granularity is finer, but the document-level and sentence-level sentiment analysis assume that the whole text and sentences contain only one kind of sentiment. This premise makes the two levels of analysis impossible to discern because of different sentiment expression objects and their corresponding polarity in texts. Specific to the product, the overall sentiment orientation of the product review text may not always be consistent with the emotional tendencies of the various aspects of the product, in which case it is necessary to identify the relationship between the feature words and the sentiment words so that more detailed level sentiment analysis is required.

The purpose of the fine-grained sentiment analysis is to extract the object attributes and their corresponding sentiment elements in the comment texts at the product feature level, so fine-grained sentiment analysis is also called attribute-based opinion mining [8]. Medhat *et al.* [11] concluded that fine-grained sentiment analysis of product reviews is generally divided into four steps: sentiment identification, product attribute selection, sentiment classification and polarity identification. Therefore, the key point of opinion mining is the feature-sentiment pair extraction. Currently, the identification of product attributes is divided into ontology-based methods [12] and nonontological methods, in other words, machine-based methods [13], [14]. The methods of sentiment word extraction are divided into dictionary-based and machine learning-based methods. Although attribute word and sentiment word extraction were studied very early, there has not yet been a unified method for extracting feature-sentiment pairs. The proposed method of Hu and Liu [10] is the earliest and most popular method for extracting feature-sentiment pairs. It is based on association rules to mine attribute words and sentiment words, and this paper will also use this method to extract the feature-sentiment pairs.

### *B. COLLABORATIVE FILTERING WITH SENTIMENT ANALYSIS*

Collaborative filtering algorithms fall into two categories: user-based collaborative filtering [1] and item-based collaborative filtering [2], [3]. In short, collaborative filtering is an algorithm that makes recommendations based on similarity and the historical buying behavior of users, so the core issue is measuring similarity and user preferences effectively. However, the similarity based on ratings only reflects the user's overall attitude toward the item, and cannot identify the specific attitude of the user to different features. The review data will occur accompanied by the user's purchasing behavior. The online review includes a detailed description of some aspects of the item, and can specifically analyze the user's sentiment attitude toward different features of the item to perform a more accurate recommendation. Therefore, ItemCF takes a content-based approach when considering user preference issues. Recent research [15] has also begun to consider this issue and replaces rating similarity by mining information from textual reviews. Content-based methods need to extract product features first because these features can be used to measure the similarity of items, and thus, what features to select becomes a problem that needs to be considered.

Currently, there is less research on collaborative filtering algorithm optimization based on sentiment factors. Leung *et al.* [16] first identified that a combination of sentiment analysis with collaborative filtering can effectively improve optimization, but regretfully it did not provide the experimental results. García-Cumbreras *et al.* [17] demonstrated that the pessimistic and optimistic sentiments of users can be used to enhance the effectiveness of collaborative filtering. Wang *et al.* [18] proposed the concept of "aspect weight", that is, all aspects of an item cannot be generalized in the recommendation process. In addition, he proposed a method of latent aspect rating analysis (LARA) using the regression model. Zha *et al.* [19] proposed a method for inferring the importance of aspects from users' existing comments, but the method cannot evaluate the aspect importance of a new comment. Zhang *et al.* [20] proposed an explicit factor model (EFM). The user preference attention matrix and the item quality matrix are used to consider the user's preference, and the recommendation result is interpretable by combining the user's favorite aspect and the quality of the item in this aspect. The limitation of the EFM model is that it cannot distinguish the different interests from different product features. Meanwhile, the accuracy of converting users' implicit feedback into an explicit evaluation needs to be further studied. Dong *et al.* [21] proposed a method for calculating sentiment benefits from user comments and proposed a recommendation strategy that combined feature sentiment with product similarity. However, multiple sets of experimental data products have similarities, and the model's scalability needs further verification. Wang and Wang [22] further considered the difference between coarse-grained and fine-grained sentiment analysis. The author believed that the more similar the user's perspective, the stronger the consistency of the user preferences. In addition, the author proposed an opinion-enhanced collaborative filtering (OECF) model measured by the degree of concern and criticism. However, this method is based on UserCF and does not take into account the characteristics of the item. Moreover, the user has different needs for different aspects of the item, which is exactly the work considered in this paper.

The above works all attempt to optimize collaborative filtering through the methods of user profile and user feature sentiment calculation, but do not consider the influence of the person's actual demand for certain aspects of the item for the recommendation effect. Recently, Bauman *et al.* [23] began research from the perspective of "the most valuable aspect" and proposed a recommendation method that not only considers the user's interest but also improves the user experience.

This paper shows that the existing research has moved from simple "like" or "dislike" to actual requirements when considering sentiment factors and the user's needs, which is similar to the feature filtering proposed in this paper. However, in general, there is still a lack of recommendation methods that include sentiment analysis based on aspect-level similarity and the user's needs. Further research is needed.

### *C. KANO MODEL*

The Kano model [24] is an effective tool for classifying and sorting users' attributes. It reflects the nonlinear relationship between product performance and customer satisfaction through the user's different sentiment attitude when a certain function exists but is missing, as shown in Fig. 1. According to the relationship between different types of quality characteristics and customer satisfaction, the quality characteristics of a product or service are divided into five categories: like, must be, neutral, live with, and dislike. When users are faced with the existence or absence of a function, they use the five attitudes to make a choice, as shown in Table 1.

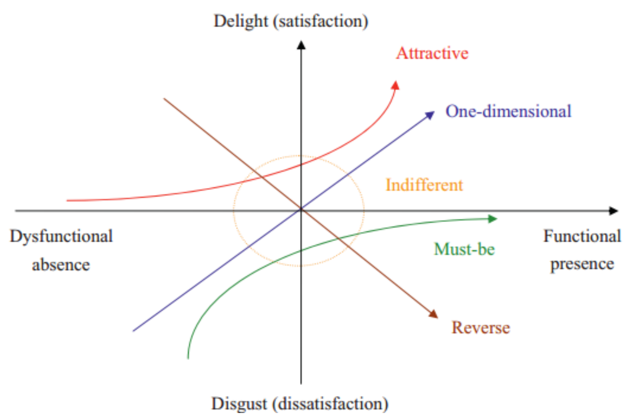



**FIGURE 1.** An illustration of the Kano model.

The evaluation results of two attitude dimensions will correspond to a certain kind of attribute. There are six possible

**TABLE 1. An illustrated questionnaire applied to the conventional Kano model.**

| | | Like | Must be | Neutral | Live with | Dislike |
|---|---|---|---|---|---|---|
| Feature | Concern | ✓ | | | | |
| m | Require | | | | ✓ | |

attributes: Must-be, One-dimensional, Attractive, Indifferent, Reverse and Questionable. The specific meaning of each attribute is explained as follows:

- Must be (M): If there is such a property, customer satisfaction will not be improved, and if it does not exist, the customer will be dissatisfied, and the user satisfaction will drop significantly.
- One-dimensional (O): The user will be satisfied, if not, the user will not be disappointed
- Attractive (A): The user will be very satisfied, and the user satisfaction will have a greatly improved emotional attitude; if not, the user will not be disappointed.
- Indifferent (I): It means that there is no need for this attribute, whether or not it exists, there will be no impact on the user experience; in other words, if it does not exist, it will not cause the customer to be dissatisfied.
- Reverse (R): If there is such a property, user satisfaction will decline.
- The Kano model also provides a scale to represent the corresponding requirements under different circumstances, as shown in Table 2:

**TABLE 2. An evaluation summary for classifying Kano categories.**

| Functional presence | Dysfunctional absence | | | | |
|---|---|---|---|---|---|
| | Like(L) | Must Be(M) | Neutral(N) | Live-with(W) | Dislike(D) |
| Like(L) | Q | A | A | A | O |
| Must Be(M) | R | I | I | I | M |
| Neutral(N) | R | I | I | I | M |
| Live-with(W) | R | I | I | I | M |
| Dislike(D) | R | R | R | R | Q |

## III. RESEARCH DESIGN

The flow chart of collaborative filtering recommendation algorithms combined with sentiment analysis and a fuzzy Kano model proposed in this paper is shown in Fig. 2; the main steps are as follows: (1) Extract feature-sentiment pairs. First, a series of data preprocessing needs to be performed on the comment data, which includes above segmentation, stemming, POS (part-of-speech) tagging, removal of stop words and extraction of feature-sentiment pairs. (2) Use the fuzzy Kano model for analysis. By combining sentiment analysis with the Kano model, different user needs with different features are analyzed. (3) Recommend products. This paper replaces the traditional similarity by calculating the

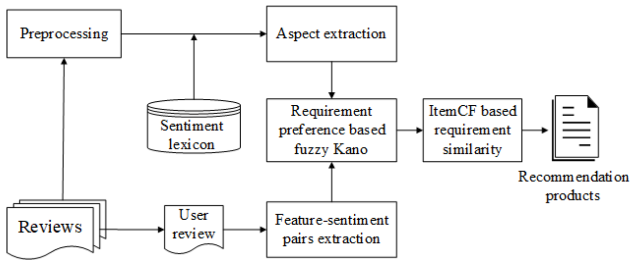



**FIGURE 2. The flow chart of our proposed research.**

similarity between items based on feature requirements and then calculates the ItemCF through the products reviewed by the user u to determine the candidate items set.

### A. EXTRACTION OF FEATURE-SENTIMENT PAIRS

This paper uses the Python NLTK package for text preprocessing. NLTK (Natural Language Toolkit) is a mature and powerful natural language processing tool which can process word segmentation, perform stemming, POS tagging, remove stop words and perform a series of work by a variety of corpora. In the data preprocessing part, each comment is divided into words with stemmed indexes, and then stop words are removed along with 17 punctuation marks from the comments, including ',', '.', ':', ';', '?', '(', ')', '[', ']', '&', '!', '∗', '@', '#', '$', '%', '…'. For the residual words, we keep words that appear in reviews that have appeared more than twice and no more than 80% and mark POS tags for these words. On the one hand, a word must be appeard at least two times in all text, which can be think is valid word. On the other hand, we set no more than 80% because if a word appears more than 80% frequently, the information content contained in the semantic information will be extremely small and cannot be considered as an aspect.Because product features are mainly composed of nouns, the words describing sentiment are mainly composed of adjectives and adverbs [8], and not all nouns appear as product features. Therefore, this paper constructs the feature-sentiment pairs by extracting the nouns with the appearance of sentiment words. Because the products between the different categories are not comparable, this paper only analyzes a single category case. According to the POS tagged in preprocessing, this paper uses the approach of Hu and Liu [8] to extract the feature-sentiment pairs, which is a more mature and popular method based on the relationship and the cooccurrence frequency between product feature words and opinion words.

First, the product features are extracted from all of the comments of the dataset, and the cooccurrence frequency of the feature words and the sentiment words in the same sentence are counted. Then, the top 30% of the single nouns are selected as feature candidates. Second, the similarity of the features is compared, and the features are dimensionality reduced [22] by combining the analogous features. In addition, the features in the feature candidate set are divided into several clusters by a clustering method, so that the features in each cluster represent a certain aspect of the product. What requires special attention is that, for the same category of products, although the number of features described is large, the described aspects are limited; the purpose of the clustering here is to classify the features effectively instead of being dedicated to the pursuit of the accuracy of the feature groupings. According to the word frequency statistics of the results in each cluster, representative N features are selected as keywords, and the aspects represented by each cluster are defined to make up the aspect-feature distribution matrix.

For the judgment of sentiment polarity, the feature is marked as +1 if the sentiment corresponding to the feature is positive, and −1 if the sentiment corresponding to the feature is negative. There is some noise in the review data, which is manifested in the opposite attitude toward the polarity and scoring. This uncertainty can have a considerable impact on research related to sentiment analysis. A feature will be considered to be noise data if the feature is marked as −1 and scored a 4.0 or 5.0 or marked as +1 with negative polarity, and will be removed from the experimental dataset.

### B. REQUIREMENT PREFERENCE ANALYSIS BASED ON FUZZY KANO

Before constructing a preference matrix, we need to use the aspect-sentiment distribution matrix as the basis. To calculate the sentiment distribution of user u on different aspects of item i, the comment data generated by the user in the purchase behavior needs to be expressed in a quaternion{user, item, aspect, sentiment} and summed to form a matrix of aspect-sentiment distribution.

Each row represents a different aspect, and each column represents a different item. The corresponding position in the matrix represents the user's attitude toward the aspect m of the n-th item. A positive attitude is recorded as 1, a negative attitude is recorded as −1, and neutral or no review is recorded as 0. There is a group of n-dimensional vector distributions for one aspect, with m groups in total.

For instance, in Table 3, the first row in the table indicates that for aspect 1, the user gives a positive evaluation for item 1, a neutral or no evaluation for item 2, and a negative evaluation for item n. So each row in the matrix represents the user's sentiment vector for that aspect. For the n-dimensional vectors, we calculate the user's degree of desire and importance for a particular feature. The degree of desire described in this article is used to express the probability of a preference when a feature has a functional presence, and the degree of importance represents the probability of dislike when a feature has dysfunctional absence, as shown in (1) and (2).

**TABLE 3.** An illustrated matrix of aspect-sentiment distribution.

| | Item 1 | Item 2 | …… | Item n |
|---|---|---|---|---|
| Aspect1 | 1 | 0 | | -1 |
| Aspect 2 | 0 | -1 | | 0 |
| …… | | | | |
| Aspect m | 1 | 0 | | 1 |

$$Love(u, A_{\rm i}) = \frac{count(u)}{m \times n} \times \frac{count(A_i^+)}{count(A_i)} \tag{1}$$

$$\text{importance}(\text{u,A}_i) = \frac{\text{count}(\text{A}_i^+)}{\text{m}} \times \frac{\sum S_{\text{A}_i}}{\sum S_u} \tag{2}$$

where $count$(u) refers to the number of sentiment attitudes (+1 or −1) given by user u for some aspect, count($A_i^+$) denotes the number of positive emotions given by u for aspect i, and count($A_i$) represents the number of reviews for aspect i. $\sum S_{A_i}$ represents the sum of all emotional attitudes toward aspect i and $\sum S_u$ refers to the sum of the sentiment scores owned by user u.

The probability distribution is divided by 20%, corresponding to the five dimensions of the Kano evaluation matrix, that is, [0, 20%] section corresponds to 'L', [20%, 40%] section corresponds to 'M', [40%, 60%] section corresponds to 'N', [60%, 80%] section corresponds to 'W', [80%, 100%] section corresponds to 'D'. Because human emotions cannot strictly be divided by intensity, this paper proposes the fuzzy Kano model. Especially when the user has different feelings toward the functional presence and dysfunctional absence at the same time, fuzzification can effectively express the user's sentiment uncertainty [25]. The proposed fuzzy Kano model turns people's Boolean choice results into probability expressions so that the user is no longer limited to just one option, as shown in Table 4.

**TABLE 4.** An illustrated matrix of aspect-sentiment distribution.

| | | LIKE | MUST BE | NEUTRAL | LIVE WITH | DISLIKE |
|---|---|---|---|---|---|---|
| ASPECT 1 | DESIRE | 80% | 20% | | | |
| | IMPORTANCE | | | 10% | 80% | 10% |
| …… | | | | | | |
| ASPECT M | DESIRE | 30% | 60% | 10% | | |
| | IMPORTANCE | | | | 30% | 70% |

Taking Feature 1 as an example, the user's degree of desire and importance can be transformed into two five-vector representations: concern = (0.8, 0.2, 0, 0, 0), require = (0, 0, 0.1, 0.8, 0.1), Using a matrix multiplication operation concern$^{\rm T}$×require, a five-row and five-column relationship matrix R is obtained as shown in (3).

$$\text{R} = \begin{pmatrix} Q & A & 0.08 & 0.64 & 0.08 \\ R & I & 0.02 & 0.16 & 0.02 \\ R & I & I & I & M \\ R & I & I & I & M \\ R & R & R & R & Q \end{pmatrix} \tag{3}$$

The Kano model can also be expressed in a two-dimensional 5x5 matrix K for different classification requirements as shown in (4):

$$K = \begin{pmatrix} Q & A & A & A & O \\ R & I & I & I & M \\ R & I & I & I & M \\ R & I & I & I & M \\ R & R & R & R & Q \end{pmatrix} \tag{4}$$

By comparing the above two matrices, the probability distribution for different requirements in the Kano model can be obtained as shown in (5):

$$\text{possibility}=\left\{\frac{0.72}{A}, \frac{0.02}{M}, \frac{0.08}{O}, \frac{0.18}{I}, \frac{0}{R}, \frac{0}{Q}\right\} \tag{5}$$

The positive degree of functional presence and the negative degree of dysfunctional absence can be measured by (6) and (7). It should be noted that although the Kano model is not good at quantifying customer preferences, the goal here is still to modify and quantify the assessment methods [26], [27] as much as possible.

$$D^{+} = \frac{A+\mathrm{O}-\mathrm{R}}{A+O+M+R+I} \tag{6}$$

$$D^{-} = \frac{\mathrm{O{+}M}-\mathrm{R}}{A+O+M+R+I} \tag{7}$$

Here, $D^+$ can be interpreted as an increase in the satisfaction coefficient, and $D^-$ is the dissatisfaction coefficient. The value of $D^+$ is usually positive, indicating that the user satisfaction will be correspondingly improved when a certain aspect of the product is improved. The closer $D^+$ is to 1, the greater the effect of improving the user satisfaction. $D^-$ is usually negative, representing that the user's satisfaction will decrease if some aspects are weakened. The closer the negative value is to -1, the faster the satisfaction declines.

The scattergram is divided into four quadrants according to the $D^+$ and $D^-$ coefficient values. The first quadrant indicates a high $D^+$ value and a high D- absolute value, the second quadrant indicates a high $D^+$ value and a low $D^-$ absolute value, the third quadrant indicates a low $D^+$ value and a low $D^-$ absolute value, and the fourth quadrant indicates a low $D^+$ value and a high $D^-$ absolute value. Attributes distributed in the first quadrant are called the one-dimensional attributes, which means that the product enhances this aspect and the user satisfaction increases. When the aspect is not improved, the user satisfaction decreases. Attributes distributed in the second quadrant are called attractive attributes, meaning that the user's satisfaction will not decrease without improving this aspect. However, there will be a great improvement in customer satisfaction when improving this aspect. Attributes distributed in the third quadrant are called indifferent attributes, meaning that there is no change in user satisfaction, regardless of whether these aspects are optimized or not. These points are features toward which users are indifferent. The distribution of attributes in the fourth quadrant, called the must be attribute, show that when the product enhances this aspect, user satisfaction will not improve, and user satisfaction will be greatly reduced if this aspect is not improved. Therefore, in measuring the similarity between items, priority should be given to the higher $D^+$ and lower $D^-$ in the second quadrant in the product distribution.

### C. IDENTIFY RECOMMENDED PRODUCTS BASED ON ItemCF

The item-based collaborative filtering algorithm is one of the most classic and well-known algorithms in recommendation systems that recommend item j that is similar to the item i purchased by the user according to the calculated similarity between the items. The similarity between the items is measured by (8), where $W_{ji}$ denotes the similarity between item j and item i, N(i) and N(j) denote the number of users of the favorite item I and j respectively:

$$W_{ji} = \frac{|N(i) \cap N(j)|}{\sqrt{|N(i)|\,|N(j)|}} \tag{8}$$

The user u's interest in item j can be calculated by formula (9). The larger the value of $\text{Interest}_{uj}$, the greater the user's preference for the recommended items. The largest K items are selected as the recommended candidate sets.

$$\text{Interest}_{uj} = \sum_{i \in N(u) \cap S(i,K)} W_{ji}{}^{*} r_{ui} \tag{9}$$

where $r_{ui}$ is the degree of user u's interest for item i. Because here it belongs to implicit feedback, we denote $r_{ui}$ w 1 and N(u) as the set of items that user u has purchased, and S(i,K) is the closest N items to item i.

In the measure of similarity, the proposed optimization method from the perspective of user requirements details the refinement of the overall level of the items toward the aspect level that can improve the degree of user satisfaction; other aspects that are not of concern to the user are ignored.

The preference P(u, i) of the user u for the item p is denoted as a preference vector $(p_{u1}, p_{u2}, \ldots p_{um})$ which represents the degree of user u's preference for aspect m in the item p. Therefore, ItemCF based on the aspect sentiment can rewrite (8) as (10):

$$W_{ij} = \frac{\sum_{i=1}^{m} \left|p_{ui} \times p_{uj}\right|}{\sum_{i=1}^{m} \left|p_{ui} - p_{uj}\right| + 1} (p_{ui}, p_{uj} \in \textit{the second quadrant}) \tag{10}$$

While each user's purchase history contributes to item similarity, those inactive users contribute more than active users, and we should reduce the weight of active users' cooccurrence products to obtain more accurate similarity, J. Breese and D. Heckerman proposed an IUF (inverse user frequency) parameter [28] to eliminate the influence of trending products, so that the similarity of items in formula (10) can be rewritten as (11):

$$W_{ij} = \frac{\sum_{u \in N(i) \cap N(j)} (1/\left(\log\left(1 + |N\left(u\right)|\right)\right))}{\sqrt{|N(i)|\,|N(j)|}} \tag{11}$$

$1/\log\left(1+|N(u)|\right)$ is the punishment of popular products.

Taking into account the impact of popular products, combined with the aspect sentiment of the similarity method, we can use formula (12) to indicate the user u for the degree of preference for item i:

$$\text{Interest}_{ui} = \frac{\frac{\sum_{i=1}^{m} |p_{ui}+p_{uj}|}{\log(1+\sum_{i=1}^{m} |p_{ui}\times p_{uj}|)}}{\sum_{i=1}^{m} \left|p_{ui} - p_{uj}\right| + 1} \tag{12}$$

## IV. EPERIMENT

### A. EXPERIMENT SETTINGS

This paper uses the Amazon Review Data dataset[1] that was presented in [29] to test the validity of the proposed algorithm. The dataset includes product review data from Amazon from May 1996 to July 2014, containing data such as ratings and review text, and according to the classification of the website, it divides the data into 24 categories according to the classification made on the website, including musical instruments, office supplies, Kindle stores and smartphones. For research purposes, this dataset provides a K-cores dataset so that there are at least K (K = 5) reviews applied to each of the remaining users and items. This paper uses the reviewerID, asin, reviewText and overall as the experimental data in the dataset. The official sample format of the dataset is as follows:

*{*
*"reviewerID": "A2SUAM1J3GNN3B",*
*"asin": "0000013714",*
*"reviewText": "I bought this for my husband who plays the piano. He is having a wonderful time playing these old hymns. The music is at times hard to read because we think the book was published for singing from more than playing from. Great purchase though!",*
*"overall": 5.0,*
*}*

To reduce the impact of the datasets, four categories are selected for the experiment in this paper. Experimental data statistics are shown in Table 5. The last two columns represent the average number of products reviewed by one user and the average number of users that reviewed one product.

**TABLE 5.** Experimental dataset information summary.

| Category | Reviews | Users | Products | Reviews/Users | Reviews/Products |
|---|---|---|---|---|---|
| Musical Instruments | 10,254 | 1,428 | 900 | 7.18 | 11.39 |
| Smartphones | 194,439 | 27,780 | 10,421 | 6.97 | 18.66 |

The experiments use the method as a real-data offline experiment. First, the experimental data are randomly divided into M (M = 10) parts, where M-1 is used to train the item clustering and user interest model as a training set, and 1 predicts the user's recommendation and calculates the corresponding evaluation index as a test set. To ensure the accuracy of the experiment, a five-fold cross-validation method was used, and the average of the five experiments was repeated to determine the specific value.

[1] http://jmcauley.ucsd.edu/data/amazon/.

### B. EVALUATION METHODOLOGY

Since the recommended algorithm in this paper belongs to a top-N recommendation, the evaluation matrix includes precision, recall, and F-value. The precision expresses the accuracy of the recommendation algorithm. The recall expresses the comprehensiveness of the recommendation system. As the recall rate increases, the precision decreases. Therefore, it is difficult to judge the recommendation system based on only one index. To consider two indexes synthetically, Wang and Hsueh [30] proposed the F-value as the harmonic average of the two. As shown in formula (13) ∼ (15):

$$precision = \frac{\sum_u |R_u \cap T_u|}{\sum_u |R_u|} \tag{13}$$

$$recall = \frac{\sum_u |R_u \cap T_u|}{\sum_u |T_u|} \tag{14}$$

$$F = \frac{precision^{*}recall^{*}2}{precision + recall} \tag{15}$$

This paper sets up two baselines to evaluate the performance of our algorithm:

**ItemCF:** a traditional item-based collaborative filtering algorithm [2].

**OECF:** The author believes that the more similar the user opinions, the stronger the consistency of the user preferences. In addition, an opinion-enhanced collaborative filtering (OECF) model measured by the degree of concern and criticism is proposed [22].

This experiment evaluates the performance of the proposed algorithm from the following three aspects: the impact of the number of aspects on the recommendation effect, the effect of different numbers of recommended products on the recommendation effect, and the recommendation performance under different algorithms. The same category of products can be explained in similar aspects, but due to the different features of the product, different categories require that the number of aspects will be different. In addition, when comparing different algorithms, the same category of products should be considered in the same aspects. This section uses ItemCF to determine the best aspects for each category of product in the experiment. Obviously, the different quantities of products lead to different impacts on the recommendation effect. In this paper, we verify the differences between the algorithm's precision ability and recall ability by increasing the recommended number of N. As a top-N recommendation, N should not be too large or too small, so the reasonable minimum N is set as 5, and the maximum is 30. In addition, the paper evaluates the recommend performance by F-value.

## V. EXPERIMENT RESULT

### A. INFLUENCE OF THE NUMBER OF ASPECTS

First, the experiment extracted product aspects. After feature extraction and incorporation, the features needed to be clustered. The effect of clustering directly affects the aspect definition and performance of the algorithm. Table 6 notes the relationship between the number of clusters K and the recommendation effect of the algorithm, where K was incremented by a multiple of two.

**TABLE 6. The relationship between the number of aspects K and ItemCF performance.**

| K | Precision (%) | Recall (%) | F value (%) |
|---|---|---|---|
| 2 | 19.88 | 37.96 | 26.09 |
| 4 | 21.78 | 38.31 | 27.77 |
| **6** | **22.35** | **41.23** | **28.99** |
| 8 | 21.30 | 40.52 | 27.92 |
| 10 | 21.29 | 39.99 | 27.79 |
| 12 | 21.18 | 39.82 | 27.65 |
| Smartphones | | | |
| K | Precision (%) | Recall (%) | F value (%) |
| 2 | 10.31 | 9.88 | 10.09 |
| 4 | 10.22 | 11.66 | 10.89 |
| 6 | 10.46 | 11.02 | 10.73 |
| **8** | **12.54** | **11.26** | **11.87** |
| 10 | 11.94 | 11.64 | 11.79 |
| 12 | 11.69 | 11.81 | 11.75 |

Apparently, the optimal cluster numbers corresponded to different datasets. The precision and F-value of the musical instruments and smartphones achieved the best effect when K = 6 and K = 8 respectively. In addition to large fluctuations in office supplies, the recall values were not all at the highest levels but were within the acceptable range. By contrasting categories from the perspective that this part of the experiment not only determines the optimal number of clusters for each category but also concludes that the more aspects of the product that need to be analyzed, the more differences in the user's preferences, accordingly, the number of clusters in smartphones is more than in music devices. It indicates that the more complex the product aspects that need to be analyzed, the more different the user's preferences may be.

### B. INFLUENCE OF THE NUMBER OF RECOMMENDED ITEMS

The method proposed in this paper belongs to the top-N recommendation and then evaluates the influence of the recommended number N to precision and recall, where N is incremented by a multiple of five.

For musical instruments, all three algorithms are optimal at N = 15, with OECF performing better than ItemCF but worse than the ASCF. ASCF precision reached 22.91%, OECF precision was 22.52%, and ItemCF precision was 22.35%. In the smartphones category, the performance of the three algorithms were above relationship, with the highest still accounting for 13.63% of the ASCF, the lowest was 12.54% of the ItemCF, and the OECF was 12.81%.

As seen in Fig. 3, at the level of accuracy, OECF can outperform ItemCF by combining with fine-grained sentiment analysis, while ASCF outperformed OECF. This shows that the proposed method of feature clustering and Kano aspects grouping can improve the recommendation precision. Moreover, ASCF achieves the optimality at N = 10, while OECF and ItemCF achieve the optimality at N = 15 and N = 20, indicating that ASCF is faster when the recommended quantity is lower, but also from another perspective that grouping can be faster at identifying the user's requirements.

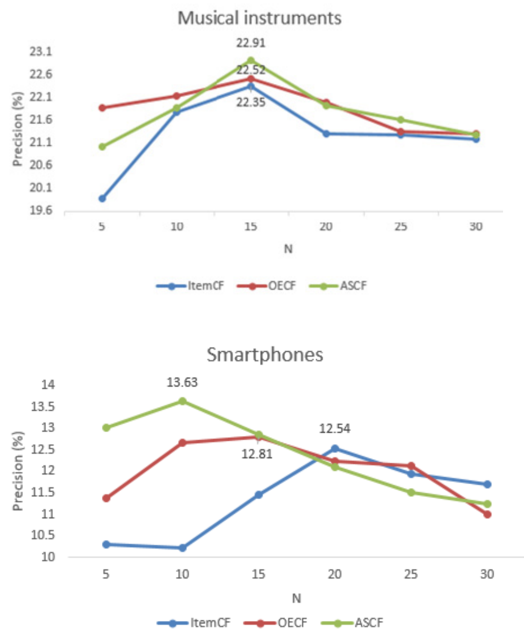


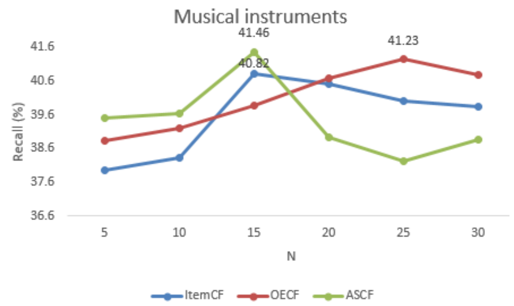

**FIGURE 3. The relationship between precision and N.**

In Fig. 4, in terms of the recall level, we can see that ASCF had the best performance and the required number of recommendations. ASCF was the best at 41.46% performance in musical instruments and reached the optimum at N = 15. OECF's recall of 41.23%, although similar with ASCF, achieved the optimal at N = 25. Although ItemCF is also optimal at N = 15, it was found to have the worst overall capacity. In the smartphone category, all three algorithms achieved the optimum at N = 25. The three algorithms maintained the above relationship, with the highest being ASCF with 12.75%, and the lowest being ItemCF at 11.81%. OECF was between the two with 12.23%.

Table 7 shows the F-values for the recommended performance of the three algorithms. As you can see, the ASCF proposed in this paper showed significant improvement over the other two methods in recommended performance, whether for musical instruments or smartphones.

### C. INSTANCE ANALYSIS

To illustrate the effectiveness of clustering, the results of the 4 most representative characteristic words were selected



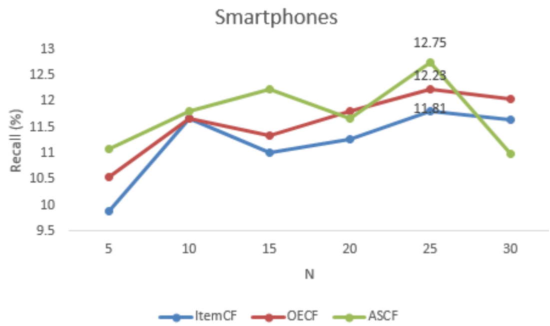




**FIGURE 4.** The relationship between recall and N.

**TABLE 7.** The comparison of F value between three algorithms.

| | ITEMCF | OECF | ASCF |
|---|---|---|---|
| MUSICAL INSTRUMENTS | 28.88 | 29.13 | 29.51 |
| SMARTPHONES | 12.16 | 12.51 | 13.18 |

from the frequency statistics in each cluster as the evaluation index and to define the aspects. Taking the relatively complex smartphone category as an example, we can see that when K = 8, each cluster clearly represented the different aspects of the product, which further illustrates the validity and rationality of the number of clusters, as shown in Table 8.

**TABLE 8.** Aspect- characteristic words distribution matrix.

| ASPECT | CHARACTERISTIC WORDS | | | |
|---|---|---|---|---|
| BATTERY LIFE | STANDB Y | TIME | POWER | BATTERY |
| SCREEN | SCREEN | LARGE-SIZE | APPEARAN CE | SIZE |
| IMAGE | PICTURE | PHOTOGRA PH | PIXEL | RESOLUTION RA TIO |
| HARDWAR E | STORAG E | HARDWARE | CPU | MEMORY |
| SOFTWARE | SYSTEM | IOS | SMART | ANDROID |
| FUNCTION AL | FEELING | KEY | TYPEFACE | VOICE |
| PRICE | PRICE | COST | MONEY | SALABLE |
| LOGISTICS | LOGISTI CS | SPEED | SERVICE | QUALITY |

In addition, in the practical application, different users have different requirements on the quantity and accuracy of recommended products. For the quantity problem, from the system product level, the output of ItemCF is a set of products sorted according to the relevance of a certain product. So in a real user system, the user will first see the most relevant products, such as 5, if they are not satisfied, they will continue to turn pages to choose from the 6th to the 10th product. When the user selects the nth product in the recommendation, n is the quantity requested by the user, which does not conflict with the research.

For the accuracy problem, we envision that you want to ask how to balance the degree of precision and recall. In terms of quantity, there are three cases: the number of recommendations is just right, or more, or less. We will not go into details about the situation just right. If there is more recommendation, that is to say, the user has found the favorite item before reaching the highest precision, there is no problem. For the third case, the recommended number does not meet the user's needs, we will be recommended according to the number of recalls.

## VI. CONCLUSION AND FUTURE WORK

The traditional collaborative filtering algorithm, as an algorithm based on a similarity measure, cannot determine whether items that are similar to the user's purchases are the items they truly like. This paper explores the views of user's reviews and calculates the similarity of items from the perspective of user's requirements, instead of the traditional similarity calculation method, to improve the recommendation precision. Experimental results show that the ASCF algorithm proposed in this paper effectively improves the accuracy of the recommendation.

Inevitably, there are some limitations to this study. The main tasks for the next step are as follows. (1) Different categories of products have different category characteristics. However, for the categories with insignificant features, whether this algorithm can effectively improve the recommendation accuracy needs further verification. (2) collaborative filtering has sparseness problems when the user's purchasing behavior has only a very small number of items; this paper does not propose a specific optimization solution for sparseness, which may have some impact on the recommendation efficiency.

In some specific dates, user maybe change their demand, which will influence our experience chonclusion. There will be special marketing activities on special dates that will affect the user sentiment to bias the Kano model. In future we will try to predict that the abnormal time point from the time dimension in the annual cycle. This factor will not be considered in the basic research. Another research concentrate on a set of data, such as price, comment, that appear at these time points. And then compare them with the results of basic research to find out whether the user's needs change and the direction of the change affects the final recommendation result during a certain period of time.

## REFERENCES


[1] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proc. ACM Conf. Comput. Supported Cooperat. Work*, 1994, pp. 175–186.

[2] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. WWW*, 2001, pp. 285–295.

[3] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Comput.*, vol. 7, no. 1, pp. 76–80, Jan./Feb. 2003.

[4] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends Inf. Retr.*, vol. 2, nos. 1–2, pp. 1–135, 2008, doi: 10.1561/15000000011.

[5] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: Sentiment classification using machine learning techniques," in *Proc. Conf. Empirical Methods Nat. Lang. Process.*, 2002, pp. 79–86.

[6] D. Jin, X. Ma, Y. Zhang, H. Abbas, and H. Yu, "Information diffusion model based on social big data," *Mobile Netw. Appl.*, vol. 23, no. 4, pp. 717–722, 2018.

[7] V. Ng, S. Dasgupta, and S. M. N. Arifin, "Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews," in *Proc. COLINGACL Main Conf. Poster Sessions*, Jul. 2006, pp. 611–618.

[8] M. Hu and B. Liu, "Mining opinion features in customer reviews," in *Proc. 19th Nat. Conf. Artif. Intell.*, 2004, pp. 755–760.

[9] B. Liu, M. Hu, and J. Cheng, "Opinion observer: Analyzing and comparing opinions on the Web" in *Proc. 14th Int. Conf. World Wide Web*, vol. 5, 2005 pp. 342–351.

[10] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, vol. 4, 2004, pp. 168–177.

[11] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Eng. J.*, vol. 5, no. 4, pp. 1093–1113, 2014.

[12] L. Liu, X. Nie, and H. Wang, "Toward a fuzzy domain sentiment ontology tree for sentiment analysis," in *Proc. 5th Int. Congr. Image Signal Process. (CISP)*, Oct. 2012, pp. 1620–1624.

[13] T. Wang, Y. Cai, H. F. Leung, R. Y. K. Lau, Q. Li, and H. Min, "Product aspect extraction supervised with online domain knowledge," *Knowl.-Based Syst.*, vol. 71, pp. 86–100, Nov. 2014.

[14] Z. Yan, M. Xing, D. Zhang, and B. Ma, "EXPRS: An extended pagerank method for product feature extraction from online consumer reviews," *Inf. Manage.*, vol. 52, no. 7, pp. 850–858, 2015.

[15] J. Xu, X. Zheng, and W. Ding, "Personalized recommendation based on reviews and ratings alleviating the sparsity problem of collaborative filtering," in *Proc. 9th IEEE Int. Conf. E-Bus. Eng. (ICEBE)*, Sep. 2012, pp. 9–16.

[16] C. W. K. Leung, S. C. F. Chan, and F. Chung, "Integrating collaborative filtering and sentiment analysis: A rating inference approach," in *Proc. ECAI Workshop Recommender Syst.*, 2006, pp. 62–66.

[17] M. García-Cumbreras, A. Montejo-Ráez, and M. C. Díaz-Galiano, "Pessimists and optimists: Improving collaborative filtering through sentiment analysis," *Expert Syst. Appl.*, vol. 40, no. 17, pp. 6758–6765, 2013.

[18] H. Wang, Y. Lu, and C. Zhai, "Latent aspect rating analysis on review text data: A rating regression approach," in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, vol. 10, 2010, pp. 783–792.

[19] Z.-J. Zha, J. Yu, J. Tang, M. Wang, and T.-S. Chua, "Product aspect ranking and its applications," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 5, pp. 1211–1224, May 2014.

[20] Y. Zhang, G. Lai, M. Zhang, Y. Zhang, Y. Liu, and S. Ma, "Explicit factor models for explainable recommendation based on phrase-level sentiment analysis," in *Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, vol. 14, 2014, pp. 83–92.

[21] R. Dong, M. P. O'Mahony, M. Schaal, K. McCarthy, and B. Smyth, "Combining similarity and sentiment in opinion mining for product recommendation," *J. Intell. Inf. Syst.*, vol. 46, no. 2, pp. 285–312, 2016.

[22] W. Wang and H. Wang, "Opinion-enhanced collaborative filtering for recommender systems through sentiment analysis," *New Rev. Hypermedia Multimed.*, vol. 21, nos. 3–4, pp. 278–300, 2015.

[23] K. Bauman, B. Liu, and A. Tuzhilin, "Aspect based recommendations: Recommending items with the most valuable aspects based on user reviews," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, vol. 17, pp. 717–725, Aug. 2017.

[24] N. Kano, "Attractive quality and must-be quality," *J. Jpn. Soc. Qual. Control*, vol. 14, pp. 39–48, Jan. 1984.

[25] Y.-C. Lee and S.-Y. Huang, "A new fuzzy concept approach for Kano's model," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 4479–4484, 2009.

[26] C. Piaszczyk, "Model based systems engineering with department of defense architectural framework," *Syst. Eng.*, vol. 14, no. 3, pp. 305–326, 2011.

[27] E. K. Delice and Z. Güngör, "A new mixed integer linear programming model for product development using quality function deployment," *Comput. Ind. Eng.*, vol. 57, no. 3, pp. 906–912, 2009.

[28] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proc. 14th Conf. Uncertain. Artif. Intell.*, 1998, pp. 43–52.

[29] R. He and J. McAuley, "Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering," in *Proc. WWW*, 2016, pp. 507–517.

[30] C.-H. Wang and O.-Z. Hsueh, "A novel approach to incorporate customer preference and perception into product configuration: A case study on smart pads," *Comput. Standards Interfaces*, vol. 35, no. 5, pp. 549–556, 2013.


**JINMING ZHANG** received the B.E. degree from the Queen Mary University of London and the B.A. degree from the Beijing University of Posts and Telecommunications in 2016. He is currently pursuing the master's degree with the School of Information and Security Engineering, Zhongnan University of Economics and Law.

**DEXIN CHEN** received the Ph.D. degree from the School of Information Management, Wuhan University, Wuhan, 2017. She is currently a Lecturer with the Faculty of Education, Hubei University. Her research interests are machine learning, context-aware computing, and educational data mining.

**MIN LU** received the Ph.D. degree in computer application from the Institute of Software, Chinese Academy of Sciences, Beijing, in 2008. She is currently a Lecturer with the School of Information and Security Engineering, Zhongnan University of Economics and Law. Her research interests are data science, machine learning, and social network analysis.