Problem Chosen	2020 MCM/ICM Summary Sheet	Team Control Number
C		2000645

Summary

For task 1, we preprocessed numerical data and text data by using distribution-based data cleaning method and probably-based word segmentation model, respectively, to make the original data set available. Firstly, we probes the original data set from the aspects of completeness, correctness and uniqueness, which are centered on users and commodities, and obtains the data distribution, the proportion of missing values and the proportion of outliers. Secondly, for numerical data, the outliers were removed, the missing values were replaced by the mean values and the calculation and processing were performed. Then, the word segmentation model based on space segmentation is used to convert sentences into word sets, and the word sets are divided into two categories: holistic description and functional description. Finally, we have a data set that can be used for further analysis, along with the most interesting features for each item.

For task 2 (a), firstly, we divided the text keywords into three categories: degree descriptors, attitude descriptors and behavior descriptors, quantifying them respectively. Then, the quantitative score based on the affective computing model is obtained by combining the numerical score and the text comment of the users. Finally, the rating is revised from the perspective of user preferences and the credibility of comments. Data measures of ratings and reviews were obtained.

For task 2 (b), firstly, we conducted time series analysis and found that the data were unbalanced and sparse in some time periods. Secondly, a time series model based on dynamic sliding window is established and the reputation is quantitatively evaluated. Finally, we use correlation analysis to obtain that the number of comments and the characteristics of products have a greater impact on reputation, while the authenticity of comments has a smaller impact on reputation.

For task 2 (c), firstly, we establish a 3d dynamic model of user-commodity-time based on sliding time window to dig out the timing characteristics of commodity evaluation. Then a collaborative filtering algorithm based on dynamic model is designed to optimize the performance of traditional collaborative filtering recommendation engine. Finally, the factors affecting the potential success or failure of the product are obtained.

For task 2 (d) firstly, we established a star correlation model based on social network, and obtained the relationship between the occurrence of different stars and the change of specific scores. Finally, we get the result of which will cause more comments.

For task 2 (e), firstly, we established ASUM model to extract users' emotion-rating distribution; Then, the rating probability of each user under positive and negative conditions is obtained. In the end, we got the comments triggered by positive emotions, most of which were 5 stars. Most comments on negative emotions are below 2 stars.

Finally, the comprehensive conclusion puts forward the improvement suggestion.

Contents

1. Introduction

1.1 background

"Data is wealth, Mining never ends." Diverse terminal devices and countless mobile users constitute this big data society. The development of the online market is undoubtedly a key object in this context. Amazon is the largest electronic device in the United States. Commercial companies provide services in all aspects including clothing, food, accommodation and transportation. At the same time, users also have the opportunity to score and evaluate purchased products, involving both quantitative and qualitative evaluation. However, how to extract useful value from massive data has always been a So helping companies use these data to gain insights into the markets in which they participate, the timing of that participation, and the potential success of product design feature choices is a top priority.

1.2 The Task at Hand

In order to realize the Amazon shopping platform's market analysis based on product ratings and reviews, we based on the provided rating and review data for microwave ovens, baby pacifiers, and hair dryers to visualize the commodity market through the following work:

- Numerical data and text data are preprocessed using distribution-based data cleaning methods and probability-based word segmentation models, respectively, to make the original data set usable.
- Establish a quantification model based on sentiment calculation to solve the quantification problem of ratings and reviews.
- Establish a reputation evaluation model based on a dynamic sliding window to solve the problem of reputation change over time. Establish a three-dimensional user-product-time dynamic model based on a sliding time window to find the factors that affect the potential success or failure of the product.
- Establish and solve the problem of whether specific stars cause more reviews. Evaluate the correlation between stars and reviews (good, poor).
- Establish and solve whether the specific text is related to scoring.

2. Model Assumptions and Notations

2.1 Assumptions and Justifications

- Assume that users comment in time after using the product (because the title does not mention when the user purchased), that is, the comments will not be weakened due to time issues;
- It is assumed that the data given by the title does not have any reviews caused by discounts or threats to ensure that the reviews are fair and open.
- Assume that the purchase users of all products are independent, there is no crossover, and there is no mutual influence.
- Assume that there are no statements in the user's text reviews that are not related to shopping attributes.

2.2 Notations

Symmols	Definition

$\deg_i$	Descriptor of degree of comment i
$atti_i$	Attitude descriptor for comment i
beh_i	Descriptive words of conduct under comment i
score_i	Text quantitative evaluation index
α	Behavior descriptor weight
β_k	Weight of the k-th feature of the product
num_rate_i	Monthly rating growth ratio
$score\_rate_i$	Number of monthly reviews
$SIM_{i,j}(t)$	comment u_i And comments u_j Similarity of interest
C_i	Aggregation coefficient of social network node i
B_k	The median of social network node k

3. Data analysis

The conversion of raw data into usable analysis data is the key to this question. The original data set contains numerical data and text data; for numerical data, data cleansing and calculation integration; for text data, the python function library is used. The NTLK word segmentation method performs word segmentation on sentences, and then classifies them according to functions and attributes; in the end, an easy-to-analyze, high-value data set can be formed.

3.1. Data detection

In order to better understand the law of the initial data, we need to conduct data exploration on the two types of data from the completeness, correctness, and uniqueness. First, remove unnecessary data, and 15-dimensional user transaction data shown in the data table Not all are useful; due to the obvious column properties of these two types of data, we use column profiling for probing.

(1) Exploration design

For the fields given in the title, review_id and product_id are not considered because they only affect a single result; the remaining fields are divided into string, numeric, and date types. No special treatment is required for numeric types, and for text types Need to be quantified.

- For numeric (int) and date (date): give the data range distribution in the form of a boxed histogram or a pie chart;
- For string: Split and quantify text, eliminate conflict anomalies. Show rich word attributes in rich text.

(2) Exploration process

From the user's perspective, the number of users participating in the evaluation greatly affects the rating of the product; whether it is rated by the platform as a vine user has an impact on the reliability of the evaluation; whether it has been purchased and then the comment has an impact on the authenticity of the evaluation. Departure, whether the service generated by the product is worth the customer's vote; whether the product evaluation star rules need to be modified in the quantitative analysis needs to be investigated and analyzed. Therefore, we start from the number of reviews, the number of vine users, the distribution of verfied purchases, the distribution of helpful votes, the star Five aspects of the rating distribution are discussed.

From the number of vine users, fig.1, the number of vine users in hair dryers, microwave

ovens, and pacifiers is a minority, so it is relatively strict to get this certification. In other words, the comments of a few people are trustworthy, and the comments of most people are more or less There are biases, so user data needs to be corrected when determining the quantitative mode.

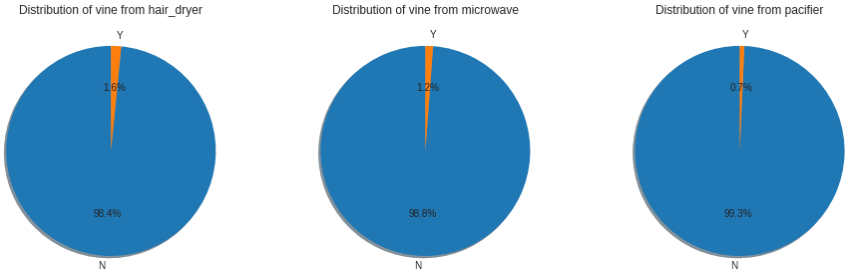

Figure 1 Distribution of vine users

From the distribution of verfied purchases, fig.2 shows that users who comment on three products without purchasing all account for more than 10%. Among them, untrue evaluations may be praises from sailors hired by sellers, or malicious comments from competitors. This greatly affects the analysis of the actual situation of the product. Therefore, to grasp the characteristics of the product from a macro perspective, we need to add a credibility factor to modify the model.

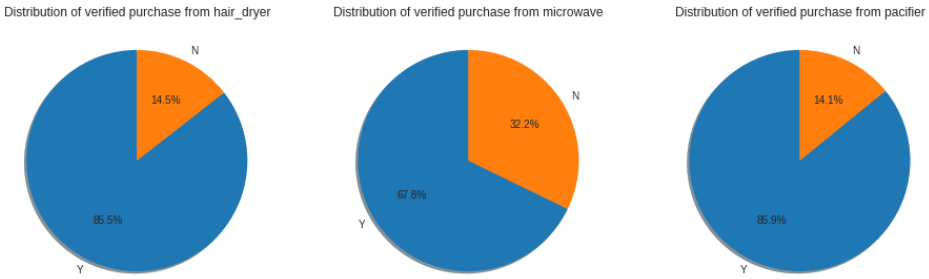

Figure 2 Authenticity distribution of the number of purchasers

Looking at fig.3 from the distribution of total vote and helpful vote, their trends are roughly the same, and the correlation is very strong. Most of them meet the actual number of votes, that is, the number of valid votes, but the actual number of votes is greater than or less than the number of valid votes, and the value of 0 is many. Therefore, when data is preprocessed, these two indicators need to be integrated first, followed by measures such as missing value processing and error value elimination.

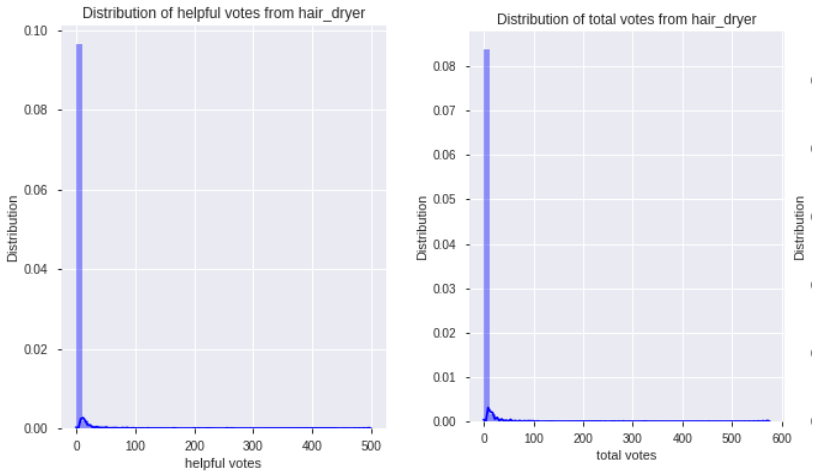

Figure 3 Distribution of useful votes

Looking at fig.4 from the star rating distribution, regardless of the above correction factors, it basically meets the rating distribution of listed products. There are some differences between different products, but it does not affect the overall analysis. Therefore, for rating, this method directly deals with the processing of ratings. Calculated using rank values.

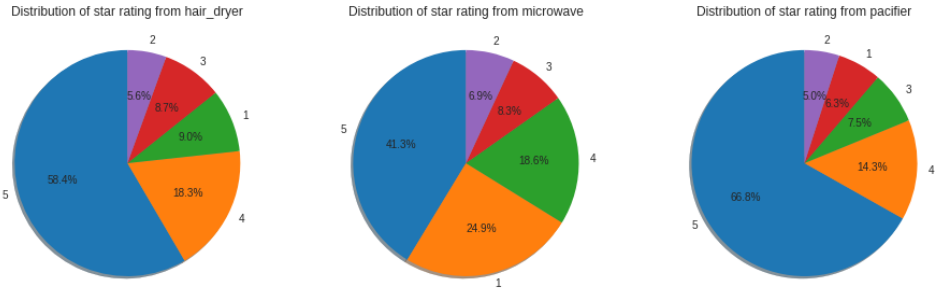

Figure 4 Rating distribution

Analyzing each of the above factors, we found that when evaluating Sunshine's star ratings and evaluating helpfulness ratings, these numerical data need to be integrated, corrected, and pre-processed to maximize the benefits of information mining.

3.2 Numerical data cleaning

Numerical data mainly include int data such as star_rating, total_votes, and date data such as review_date. Data cleaning mainly deals with the processing of various abnormal conditions on the data value. According to the results of discussion in 3.1.1, all numerical data is cleaned, including Three operations: missing value processing, error value elimination, and calculation integration.

(1) Missing value processing

For the outliers in the data table, they account for a small part of the entire data but are likely to cause large deviations in the results. This article uses the average substitution method to optimize the outliers and reduce the impact of noise.

(2) Outlier rejection

Aiming at the problem of data errors and excessive 0 values in the total number of votes and valid votes, as these are very rare special cases, this article directly deletes the entire data,

and ensures the accuracy of the analysis without affecting the overall evaluation of the product. .
(3) Computational integration

For helpful_votes and total_votes, the ratio of helpful_votes / total_votes is two-in-one. At the same time, records with too large total_votes values are filtered to avoid the problem of large numbers caused by the exposure of a certain comment.

3.3 Text data preprocessing

Considering the validity of the data, the text-type data mainly includes the review_headline and review_body string types and the vine and verify_purchase character types. For string types, first use NTLK to convert the sentence into a combination of words, then classify the words and remove the text at the same time Conflict. For character type, perform bool conversion.

(1) Text conflict elimination

First, if the overall description of the comment title conflicts with the overall description in the comment, it is considered as invalid data processing, and the entire data is deleted to ensure the effectiveness of the next text analysis.

(2) Word segmentation model based on ntlk

The NLTK function package in the python function library has great advantages in English text processing. It comes with a dictionary called dict.txt, which contains more than 20,000 words, including the number of occurrences and the part of speech. By segmentation Spaces and other methods are used to segment words. Then, after the case detection, the words are unified into lowercase. Finally, the part of speech is restored, and the words in different tenses and singular and plural forms are converted into the initial state and saved in the form of a list.

For this solution, the words are divided into two categories, one is the overall description of the product; the other is the functional keywords and description; thus two lexicons of each product are obtained: one is the overall description Thesaurus. The other is a functional keyword library.

(3) Phrase classification

For this solution, the words are divided into two categories, one is the overall description of the product; the other is the functional keywords and description; thus two lexicons of each product are obtained: one is the overall description Thesaurus. The other is a functional keyword library.

For the functional keyword database, according to the business 28 principle, which is the Palestine specific law: 80% of the total result is determined by the key 20%, and the top 20% of the main functions are selected as the representative. In this way, the original long character text is initially transformed into Quantitative descriptions, such as: keywords for the title (good, bad), keywords for the review (good, recommended, etc.), evaluation of function 1 (good), evaluation of function n (good), etc.

(4) Single-character variable processing

Vine: After the customer made accurate and insightful comments, Amazon invited him to become "Amazon Vine Voices". Therefore, this solution fully trusts the reviews of vine customers, and needs to remain skeptical of non-vine customers. Therefore, when the model was established, it was specified that:

$$\text{vine} = \begin{cases} 0, & \text{vine} \\ 1, & \text{not vine} \end{cases}$$

verify_purchase: Whether it belongs to customers who have actually purchased the product and then commented, stipulated

$$\text{verify\_purchase} = \begin{cases} 1, & \text{yes} \\ 0, & \text{no} \end{cases}$$

3.4 Model Calculation and Result Analysis

The text-based data in the table is substituted, and text conflicts are eliminated. Based on the python3.6 development environment, the Jieba function package is used to segment the words. According to the number of words, the word cloud map is generated (taking hair_dryer_review_body as an example, the rest is shown in the attachment).

Figure 5 hair_dryer_review_body_wordcloud

According to the analysis of the cloud map, it can be found that (1) the probability of occurrence of functional descriptors such as "blow", "dryer", "use", etc. is very suitable for separate isolation; (2) overall descriptors such as "great" "well" " "problem" etc., there is a clear bipolar trend, which provides a basis for the quantification of words.

After processing 3.1.2 and 3.1.3 for numerical data and text data, an analysis data set is formed, as shown in table1 (see the appendix for completeness).

Table 1 Analysis data table

Index	helpful_votes	total_votes	vine	helpful_rate	star_rating	text
1	0	0	0	0	4	dislike
2	0	0	0	0	3	Have
3	0	0	0	0	1	Heating
4	0	0	0	0	3	Favorite
5	0	0	1	0	5	Little
6	44	49	0	0.9	5	Awesome
7	32	34	0	0	1	Dryer
8	1	1	0	1	5	blowdryer

ndex stands for the id of the review, helpful votes is the number of other users' comments on the review, total votes is the total number of votes that other users have for the review, helpful rate is the ratio of the helpful rate to the total votes, and Vine consists of the text "Y"

And "N" becomes 0 and 1, and text is the text obtained by review after text processing.

4. Scoring quantitative model based on sentiment calculation

The key to this question is the quantification of textual data and the determination of product evaluation index values. First, we divide the keywords separated above into three categories: degree descriptors, attitude descriptors, and behavior descriptors, and then quantify them separately. The evaluation index value of each record is obtained by using an evaluation model based on sentiment calculation. Finally, according to the results of data exploration, the index value formula is modified to obtain data metrics for ratings and reviews.

4.1 Text keyword quantification

According to 3.4, we have proposed the text keywords, and now the data in the table is only the overall descriptive words of the product (because the first n of the functional text have been selected as the n columns). Next, we need to process the qualitative words For specific values.

Count the number of occurrences of keywords and divide them into the following three categories: (1) 'somewhat' equal degree descriptors; (2) 'good and bad' attitude descriptors; (3) 'recommended / not recommended' behavioral descriptions As the user shifts from language to action, a certain performance of the product triggers the user's further behavior, so the behavioral descriptors have the largest weight, and this voice has the desire to promote and promote other users not to buy / purchase.

Then quantify separately, the degree description word is assigned a decimal of 0.5, 0.2, etc., is determined according to the distribution, the attitude description word is assigned a good value of + 1 / -1, and the behavior description word is assigned a recommendation of 1 and a recommendation of 0 is not recommended;

4.2 Affective Computing

For the title review_headline, it is regarded as the overall evaluation of the user, and the attitude descriptor can be determined. For the text review_body, it is regarded as the detailed evaluation of the user, and the degree descriptor and the behavior descriptor can be determined.

The aggregation of customer perspectives shows the qualitative evaluation results of product characteristics. It includes the customer's service experience and subjective feelings. Therefore, this article proposes the emotional credibility index of reviews to calculate the emotional credibility of product reviews (in the First, the product recommendation model based on dynamic window extraction feature point pairs in the comments). Define the emotional credibility calculation formula of the k-th function of the product corresponding to the i-th comment as:

$$\mathrm{fun}_{i-k} = \mathrm{deg}_i * \mathrm{atti}_i + \alpha * beh_i$$

Among them, star_i is the first star rating, deg_i is the degree descriptor, atti_i is the attitude descriptor, beh_i is the behavior descriptor, fun_{i-k} is the k-th functional score of the product, and α is the weight of the behavior descriptor. The default value is 0.5.

So the tex_score_i calculation formula for the total text review is:

$$\mathrm{tex\_score}_i = \mathrm{deg}_i * \mathrm{atti}_i + \alpha * \mathrm{beh\_i} + \sum \beta_k * \mathrm{fun}_{i-k}$$

among them,β_kIs the weight of the k-th feature of the product corresponding to i reviews, taking the statistical value of the feature.

Finally, according to the 3.1.1 analysis, you can directly add the customer's star rating in the review to get the score_i, a quantitative evaluation index for each record:

$$\text{score}_i = \text{tex\_score}_i + \text{star}_i$$

4.3 Model amendments

(1) Correction of user preferences

Because each product has its own targeted service users, and there are differences in service acceptance between different users, it is important to modify user preferences.

First, divide users into three categories: easy-to-satisfaction, fairness, and non-satisfaction, and then make corrections; then, determine whether users have made multiple purchases, and appropriately reduce the weight of users who have made multiple purchases to avoid the evaluation of the product because Excessive repurchase results in a falsely high score; finally, adjust the user evaluation weight based on whether the user is a vine voice and whether they have actually purchased it.

(2) Reliability correction

Considering the subjectivity of voting and the uncontrollability of reviews, the reviews of all products are definitely not completely true. Therefore, this article adds a credibility factor γ to correct the final result to ensure the accuracy and reality of the final quantitative model.

4.4 Model Calculation and Result Analysis

Establish an sentiment analysis model for the review text and the rate star, and select the sentiment words with a frequency of top20 to make a radar chart fig. 6.

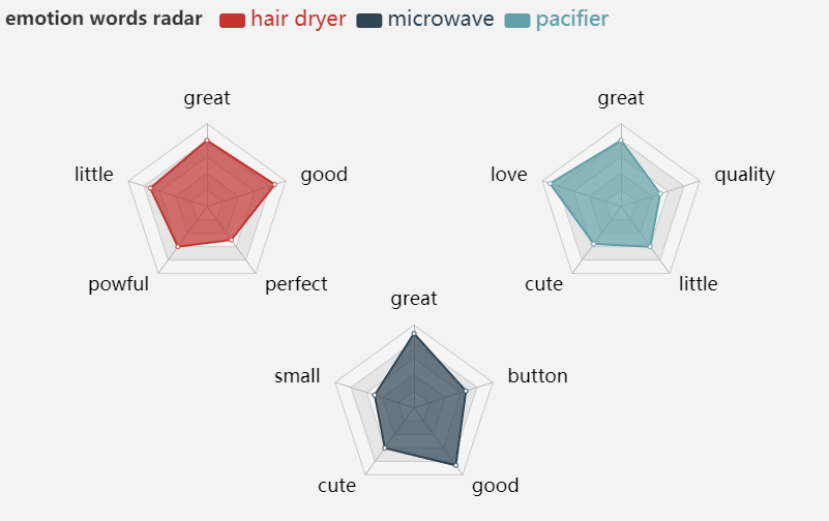

Figure 6 Radar chart for sentiment analysis

Each corner of the radar represents the attribute of the product and its emotional polarity, and the coverage represents the value of the attribute and its emotional strength. It comprehensively represents the performance of different products in the market. Analysis shows that: (1) hairdryer evaluation tends to be diverse The "little" attribute has the most influence on the reviews; (2) There are many reviews of microwave, and the review language is slightly single, of which the "button" function has the greatest influence; (3) pacifier and

microwave are similar Among them, the "quality" and "cute" properties have a greater impact on it.

Using the quantitative calculation formula in 3.2.2, α is 0.5. Put the same product evaluation together, and the results are shown in table 2.

Table 2 Results calculation table

	score	degree	behavior	attribue	function1	function2	function3
0	5.95	0.5	0.7	5.95	0.9	0.5	0.9
1	4.81	0.5	0.3	4.81	1	0.7	0.8
2	5.986	0.8	0.6	5.986	0.6	0.6	0.6
3	6.15	0.8	0.7	6.15	0.9	0.7	0.9
4	4.808	0.9	0.8	4.808	0.4	0.9	0.8
5	6.314	0.8	0.7	6.314	0.7	0.8	1
6	1.22	-0.9	-0.8	1.22	0.4	0.4	0.4
7	3.358	0.3	0.1	3.358	0.4	1	0.7
8	6.158	1	0.6	6.158	0.9	0.6	0.8
9	6.152	0.7	0.6	6.152	0.8	0.5	0.9

Among them, Score is the score finally calculated by the formula, degree is the weight of the degree descriptor of the product, behavior is the weight of the behavior descriptor of the product, attribute is the weight of the attitude descriptor, and functions1-5 are the users' scores for the product. The evaluation score of each function can be obtained by the user's feedback on the overall function of the product through the weight ratio of each function.

5. Time series model based on dynamic sliding window

Due to the unstable number of reviews on a certain day in the data set, we built a time series model based on a dynamic sliding window to analyze the reputation fluctuation of the product in the market during the time when the number of reviews was stable.

5.1 Time series analysis

First of all, the star-level change (targeted to a certain product) is calculated by year. See fig. 7. The pictures show that the sales and praise rates of the three products are on the rise, but it is not difficult to see that there are also small twists and turns of the range.

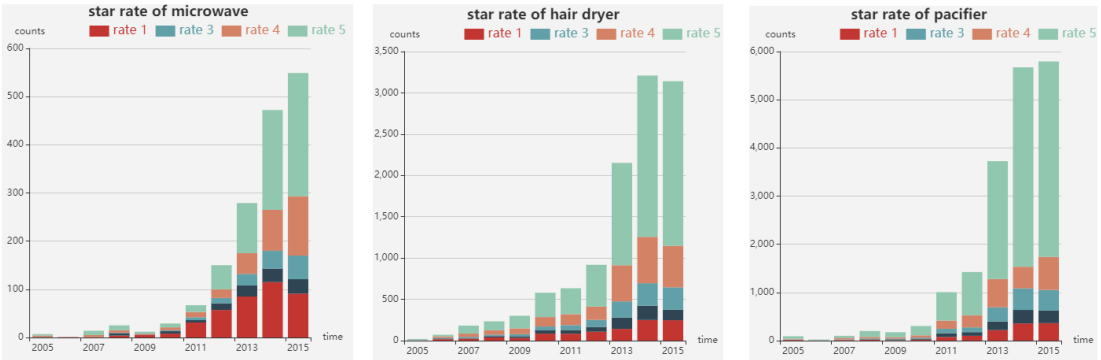

Figure 7 Star classification

In order to further study the causes of reputation fluctuations, we perform statistics on a monthly basis, such as fig.7. It can be seen that there are too many or too few comments on a

certain day, and the positive and negative comments in the comments are not uniform. There is no theoretical guidance for reputation analysis. Therefore, we establish a time series model based on a dynamic time window.

5.2 Dynamic time window design

A dynamic time window, as shown in fig.8 (where the value in the green box represents the number of comments on a day, and the gray box is the sliding window), with x number of comments (values need to be discussed) as the unit, in the time axis Swipe up. This will have a different time window every day (similar to the time can be divided into 123, 23, 345, 6), to ensure that each window has a similar number of comment values. This can exclude when the number of comments is large The effects of offsetting bad factors, sparse data / small number of comments / accidental fluctuations, and finally accumulating scores to get the average.

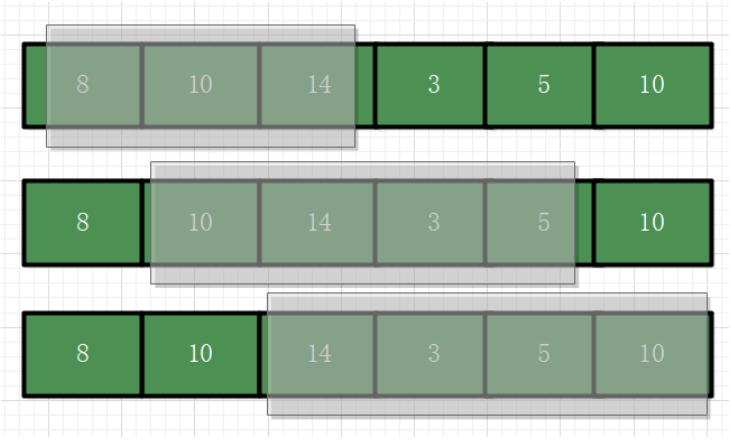

Figure 8 sliding window example

5.3 Correlation analysis between the number of product reviews and product reputation

The reputation of a product is determined using the value of the quantitative index of reviews obtained in 3.2, and the average value of the scores of all reviews of the product in each month is calculated in units of months. In order to reflect the changes in the reputation of the products, we use a differential comparison method to obtain each Monthly rating ratio, i.e .:

$$score\_rate_i = {score_{i+1}}/{score_i}$$

Similarly, the ratio of the number of comments per month is:

$$num\_rate_i = {num_{i+1}}/{num_i}$$

Use the sliding window in 3.3.2 to collect the number of reviews in units of months. Through experiments, set the size of the sliding window (the number of stable comments determined): hair_drye-104, microwave-14, pacifier-172, and get the results such as fig. Shown in 9.

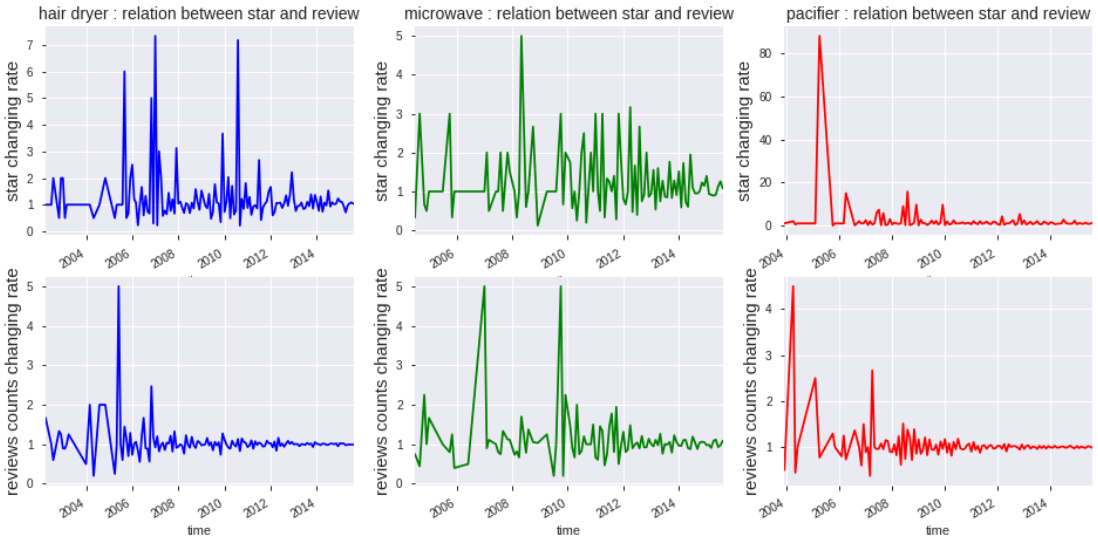

Figure 9 The relationship between the number of reviews and the average evaluation

According to the figure, the following conclusions can be drawn:

- The number of reviews is basically positively related to the average score of the evaluation. For example, the sales and rating scores of hair_dryer from 2006 to 2008 have both increased and decreased, and this phenomenon has also occurred in other products;
- The more the number of reviews, the higher the star rating of the product, that is, when the quality of the product is recognized by the user, it will bring many praises, which will once again promote the sales of the product, and increase the sales volume. At the same time, the increased sales volume It will also turn into a favorable comment on the product. This virtuous circle will bring a substantial increase in the reputation of the product, which is also in line with the actual situation;
- The fewer the number of reviews, the star rating of the product will decrease, that is, when the quality of the product is not satisfactory, the increase in bad reviews will lead to a decrease in sales.In such a vicious circle, the reputation of the product will decline.

5.4 Correlation analysis of other factors and product reputation

In this part, we will include the five good attributes before a review: good_reviews (above 2 stars), bad reviews (below 2 stars), review feature character, overall review value of the text, whole_text, and reviews whether the user is truly verified_purchase. Using heat map analysis, discuss their correlation with star_rating and get the results, as shown in fig.10.

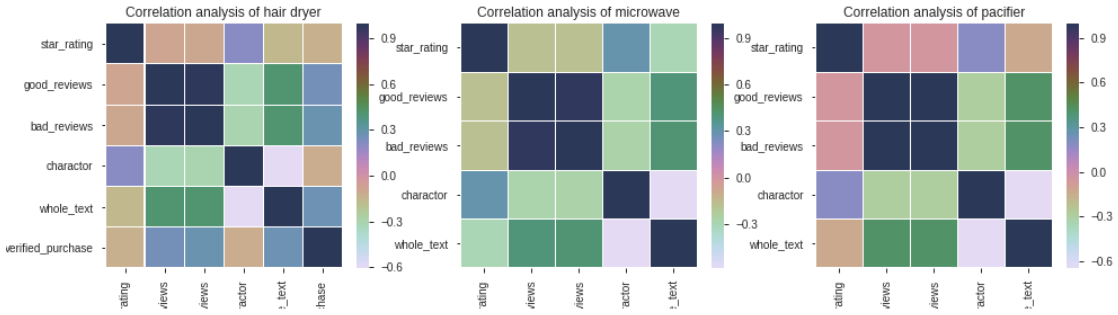

Figure 10 Correlation analysis heat map

It can be seen from the analysis chart that the evaluation star rating has the most correlation

with the characteristics of the product itself; the number of positive and negative evaluations has a considerable impact on the reputation of the product, and both maintain a high level; the authenticity of the reviewing users and the overall evaluation of the text on the product reputation The impact is small and does not play a decisive role.

In summary, the number of product reviews has the greatest impact on the reputation of a product, and the rise and fall of reputation can be predicted by monitoring the sales of the product; secondly, the characteristics of the product itself have a greater impact on reputation, and the quality of the product guarantees the continued rise of the reputation; finally, Users' individual actions, users are independent and uncontrollable, and have less effect on the reputation of the commodity market.

6. Three-dimensional user-commodity-time dynamic model based on sliding time window

Based on the analysis of Question 3, we found that factors such as the number of reviews, the characteristics of the product itself, and the authenticity of the review all have an impact on the market reputation of the product. In order to tap the data measures of the potential success and failure of the product, we use collaboration based on user evaluation. The filtering algorithm calculates the interest degree vector of different factors for the reputation of the product by establishing a three-dimensional dynamic model of user-commodity-time, thereby obtaining the most relevant influencing factors.

6.1 User-Commodity-Time 3D Dynamic Model Based on Sliding Time Window

In order to dig out the time series characteristics of product evaluation and optimize the performance of the traditional collaborative filtering recommendation engine, this paper proposes a user-product-time (hereinafter referred to as uct) three-dimensional dynamic model based on sliding time windows. This model is based on the traditional The user-product two-dimensional model adds a time dimension, and assigns a forgetting weight to the time axis.The user behavior in a time period that is closer to the current moment has a greater impact on the final recommendation result, giving a larger weight and giving The weight of user behavior in a time period far from the current moment is small, and the score information in the time far away from the current moment has little effect on the final recommendation result and can be ignored. In addition, it is necessary to consider that the user's interest has faded over time. , The user's historical ratings show a decaying trend towards the mean value, until they are refreshed by the user's recent time rating.

The three-dimensional UCT model constructed based on the above ideas is shown in Figure 11, where the X-axis represents time, the Y-axis represents the user group, and the Z-axis represents the product review group. The time window T_1-T_s is intercepted in sequence from the origin on the time axis. The window represents a time series, T_1 is the time window furthest from the current moment, T_s is the time window closest to the current moment, and each time window weight is represented by T_i, where T1 <T2 <... <Ts. On the user axis A closed plane is intercepted at a certain point and perpendicular to the user axis, as shown by the dotted rectangular box in Fig. 2.It records the scoring information of a certain user on all items in the T1-Ts time. Each registered user has such a closed plane. As time progresses, the time window scrolls forward, and these closed planes that record the user's ratings are shifted

forward at the same time.The user's rating information also gradually decays to the mean as the time window passes, so the collaborative filtering recommendation based on the UIT 3D model The engine is dynamically timed, and iteratively updates in a rolling time window.

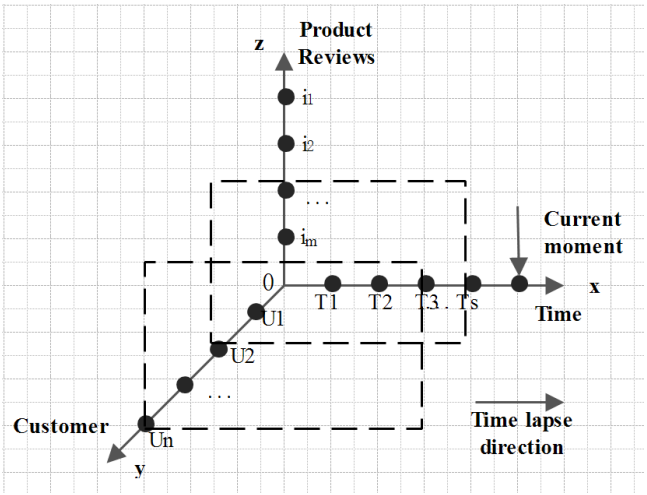

Figure 11 3D model based on user-commodity-time

6.2 Design of collaborative filtering algorithm based on dynamic model

(1) The core idea of the algorithm

Define the number of time windows as s, represented by T_1, T_2, ..., T_S; the number of users is n, represented by u_1, u_2, ..., u_n; the number of comments is m, represented by i_1, i_2, ..., i_m. In the dynamic model Find n closed planes, map them into n item-time window two-dimensional interest degree matrices, and determine the similarity between user reviews and product ratings by calculating the distance between these n two-dimensional matrices. Finally, you can find the nearest K neighbors of each product, that is, K user reviews with the highest similarity to yourself, and finally determine the determinants of the market effectiveness of the target product based on the common interest vector of these K reviews. When the time window is scrolled once You just need to translate the item-time window two-dimensional interest degree matrix as a column and update the nearest column, then use the incremental algorithm to calculate the similarity after rolling and get the recommended results.

(2) Specific steps of the algorithm

According to the core idea of the above recommended algorithm, the algorithm is divided into the following steps:

Step 1: Format user rating data

It is known that the score of the quantitative index of product reviews is calculated in 3.2.4. The score needs to be formatted into a closed interval, such as score $\in$ [1, 5], where 1 is the least interested and 5 is the most interested.

Step 2: calculate the similarity between user reviews

The interest similarity between users u_i and u_j is reflected in the distance between the two interest scoring matrices. The distance between the matrices can be indirectly obtained by calculating the distance between the corresponding column vectors of the two matrices. The components are closely related to their respective time windows, so the time window weights need to be considered when calculating similarity. Therefore, the SIM_i_j (t) formula for calculating the interest similarity between the user ui and the user uj at the current moment is

as follows:

$$\mathrm{SIM}_{i,j}(t)=\frac{\sum_{k=1}^{s} T_k \cdot \mathrm{sim}_{i,j}^{(T_k)}}{\sum_{k=1}^{s} T_k}$$

Step 3: interest similarity incremental algorithm

Considering the scalability of the algorithm, we simplify the calculation of user similarity. Using the above-mentioned time series of interest similarity, the following incremental formula is derived:

$$\begin{aligned}\mathrm{SIM}_{i,j}(t+T)&=\frac{\sum_{k=1}^{s-1} T_k \cdot \mathrm{sim}_{i,j}^{(T_{k+1})}+T_s \cdot \mathrm{sim}_{i,j}^{*}}{\sum_{k=1}^{s} T_k}\\&=e^{-\frac{\lambda}{s}}\mathrm{SIM}_{i,j}(t)+\frac{1}{\sum_{k=1}^{s} T_k}\left(\mathrm{sim}_{i,j}^{*}-e^{-\lambda}\mathrm{sim}_{i,j}^{(T_1)}\right)\\&\approx e^{-\frac{\lambda}{s}}\mathrm{SIM}_{i,j}(t)+\frac{1}{\sum_{k=1}^{s} T_k}\mathrm{sim}_{i,j}^{*}\end{aligned}$$

Step 4: recommendation decision

According to the algorithm of Step 3, the interest similarity between the two pairs of user reviews is obtained, and finally the similarity matrix of the product review group can be obtained. The similarity matrix is defined as a symmetric square matrix that records the interest similarity between all user reviews. for:

$$S=\begin{array}{c} \begin{matrix} u_1 & u_2 & \cdots & u_n \end{matrix} \\ \begin{pmatrix} 0 & \mathrm{SIM}_{12} & \cdots & \mathrm{SIM}_{1n} \\ \mathrm{SIM}_{12} & 0 & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \mathrm{SIM}_{1n} & \cdots & \cdots & 0 \end{pmatrix}_{n\times n} \end{array} \begin{matrix} u_1 \\ u_2 \\ \cdots \\ u_n \end{matrix}$$

The recommendation strategy based on the obtained interest similarity matrix is as follows:

(1) When the target user reviews u_x∈ (u_1, u_2, ..., u_n), directly find the largest K interest similarities in the u_x rows of the matrix S, and the corresponding column elements are the K users of the target user u_x Nearest neighbor user set.

(2) When the target user comment u_x does not belong to {u_1, u_2, ..., u_n}, the recommendation system encounters the user cold start problem [6]. We obtain the interest similarities SIM_x1, SIN_x2, ..., SIM_xm respectively according to the above algorithm. And then find the largest K values from it, and the corresponding user is the nearest neighbor user set of the target user u_x.

After finding the set of k nearest neighbor user reviews for the target product, use the following formula to obtain the k-neighbor user review interest vector for all product reputations, and then find the n factors with the highest interest as the image product market sales Key factors.

$$P_x = \frac{\sum_{i=1}^{K} \mathrm{SIM}_{xi} A_i^{(s)}}{\sum_{i=1}^{K} \mathrm{SIM}_{xi}} = \frac{\sum_{i=1}^{K} \mathrm{SIM}_{xi} \begin{pmatrix} w_{1s}^{(i)} \\ w_{2s}^{(i)} \\ \cdots \\ w_{ms}^{(i)} \end{pmatrix}}{\sum_{i=1}^{K} \mathrm{SIM}_{xi}}$$

6.3 Model Calculation and Result Analysis

Substituting relevant data into the result is shown in Figure 12.

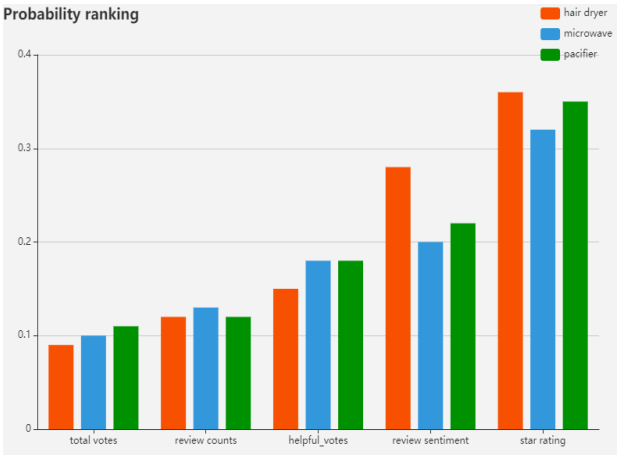

Figure 12 Product importance ranking

The figure describes the left-to-right probabilistic ordering of the importance of each element after key element analysis and collaborative filtering. The three legends represent the three products. Here are a total of 5 elements, the key from small to small. The big differences are, total votes, review counts, helpful votes, review sentiment, star rating.

7. Star network correlation model based on social network

The key to this question is to analyze the changes in the number and quality of product reviews under a specific star rating. Using the idea of social networks, by enumerating all the circumstances of a specific star rating and star rating change, digging the social relationship between different star ratings to determine Relevance of stars to reviews.

7.1 Star network correlation model based on social networks

In order to analyze the correlation between different stars, we use the comprehensive correlation coefficient method to obtain the correlation degree of each star. The principle of the comprehensive correlation coefficient algorithm to solve the correlation coefficient is as follows:

- Calculation of correlation coefficient r

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

In general, the value of r is between (-1,1), and the degree of correlation can be divided into the following cases: When $|r| \geq 0.8$, it can be regarded as highly correlated; $0.5 \leq |r$

| <0.8 , It is regarded as moderate correlation; when 0.3 ≤ | r | <0.5, it is considered as low correlation; when | r | <0.3, it indicates that the degree of correlation between the two variables is extremely weak and can be regarded as a nonlinear correlation.

- Improved calculation of absolute grey correlation coefficient

*S*tep1: Set the system's behavior order:

$$X_0 = (x_0(1), x_0(2), \cdots, x_0(m))$$
$$X_1 = (x_1(1), x_1(2), \cdots, x_1(m))$$
$$\cdots\cdots$$
$$X_i = (x_i(1), x_i(2), \cdots, x_i(m))$$
$$\cdots\cdots$$
$$X_n = (x_n(1), x_n(2), \cdots, x_n(m))$$

*S*tep2: Calculate the correlation coefficient:

$$\gamma_{0i}(k) = \frac{c}{c + \tan\left(\frac{\beta_{0i}(k)}{2}\right)}$$

*S*tep3: Calculate the application weight Woi (k):

$$W_{0i}(k) = 1 - \frac{|x_i(k) - x_0(k)|}{\sum_{k=1}^{m} |x_i(k) - x_0(k)|}$$

*S*tep4: correlation calculation:

$$\gamma_{0i} = \frac{1}{m-1} \sum_{k=1}^{m-1} W_{0i}(k) \gamma_{0i}(k)$$

- Calculation of comprehensive correlation coefficient

$$p = \frac{|\gamma_{0i}| + |r|}{2}$$

Finally, we use the correlation coefficient as the edge weight to get the network structure.

7.2 Analysis of results

In order to perform a visual analysis of the impact of reviews caused by ratings, we analyze the relationship between the appearance of different stars and changes in specific ratings based on the time series, based on the idea of social networks. First, suppose that each user's comment is Have a clear attitude and guarantee authenticity; secondly, the changes between the levels will directly affect the quality of future reviews. Use pajek software to graphically illustrate the trend of star changes, as shown in fig.13.

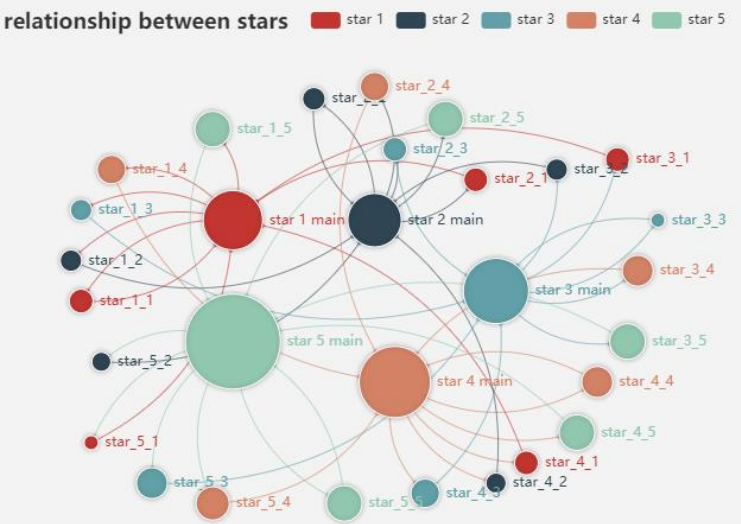

Figure 13 Rating relationship diagram

This graph shows the relationship between the appearance of different star ratings and the change of a specific star rating. Each child node represents a statistical distribution of different scores over a period of time, and each large node represents a comprehensive distribution of scores. The edge points indicate that after the current node event occurs The frequency of occurrence of each score in the next time. By analyzing fig.13, the following conclusions are drawn:

- In the social network, each star does not exist separately. Among them, 5 stars have the largest impact on the change of stars in the network, followed by 1 star, 2 stars, 3 stars, and 4 stars. This shows that the specific Positive or negative reviews can bring huge changes in reviews.
- For products with low star reviews, users are more likely to write medium stars (2 stars, 3 stars), and there are a small number of possible high star reviews. This reflects the diversity of customer groups and is in line with actual conditions;
- For products with medium-star reviews, the reviews written by users are also basically medium-star. The quality of the products itself is in a medium position, and users do not have problems of preference and disgust, which is in line with actual conditions.

8. User emotion-rating distribution extraction model

In order to classify sentiment keywords, we use the asum model to extract the user's sentiment-rating distribution and obtain the rating probability of each user in positive and negative. The results show that star reviews are positively correlated with text reviews.

8.1 User emotion-rating distribution extraction model

For the previously divided words, this article uses the asum model to extract the user's sentiment-rating distribution. First, all the comments of each user who has been preprocessed by the review in the experimental data set are integrated together as the user's corpus, and then all users' The corpus is integrated together and becomes the corpus of the asum model.

The ASUM model can use a prior sentiment dictionary to improve the accuracy of sentiment classification. In this article, when the ASUM model is set, the number of emotions is set to 2, where 0 represents positive emotions and 1 represents negative emotions. The

sentiment dictionary used is from China Knowledge Network How Net sentiment dictionary. Since the review text comes from the Internet, some network terms have been manually added to the sentiment dictionary. Table 3 shows some examples of the content of the sentiment dictionary used in this article.

Table 3 Part of sentiments words in the sentiment lexicons

Positive dictionary	Negative dictionary
Enthusiasm, satisfaction, genuine, happy, love, praise, value, praise, outstanding, cute, useful, first-rate, concession, beautiful, surprise ...	Bad, disappointed, ugly, defect, regret, bad, rise, junk, false, useless, bad, return, complaint, fooled, old, miscellaneous ...

The algorithm uses the asum model for comment mining, which can obtain the rating probability of each user under two emotions. In order to facilitate the display, in the article, only the probability value of five stars corresponding to each emotion is required to analyze Relate the emotion class descriptors and ratings.

8.2 Result Analysis

According to the user emotion-rating distribution extraction model, the user's rating distribution under positive and negative emotions is obtained, as shown in fig.14. This figure describes the relationship between the emotional tendency of reviews and the rating before the rating. Positive or negative. The outer circle represents the proportion of each rating under this emotional tendency.

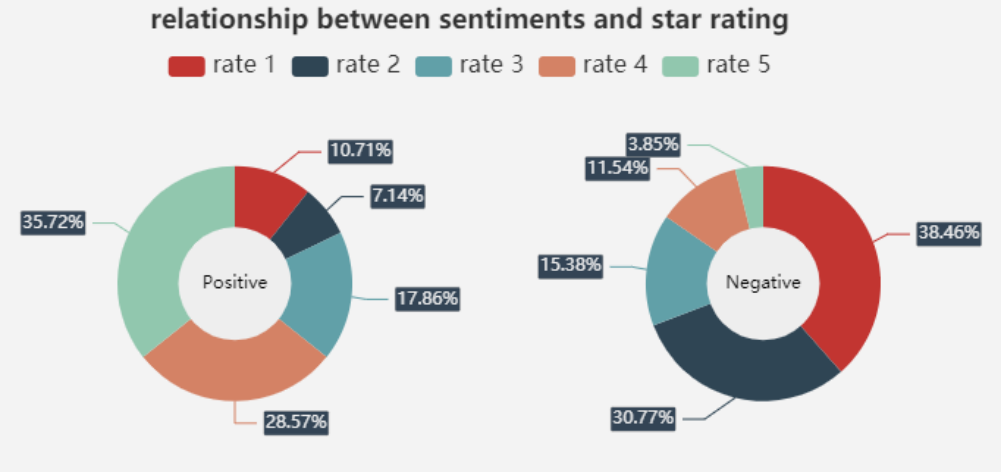

Figure 14 User sentiment-rating distribution

By analyzing fig.14, we can draw the following conclusions:

（1） Star reviews are more than 80% relevant to text reviews, and there are less than 20% anomalies, which are consistent with the user authenticity correction discussed earlier in the article.

（2） Most of the comments caused by positive emotions remain above 3 stars, with 5-star reviews accounting for the majority. The appearance of low-star reviews indicates that the ratings conflict with the reviews and can be ignored during analysis.

（3） Most of the comments sent by negative emotions are kept below 2 stars, and there are also a few high-star reviews. Records that conflict with ratings and comments

can be considered as wrong data and deleted without affecting the final analysis results .

9. Strength and Weakness of Our Model

9.1 Advantage

- Make full use of data visualization technology, with the help of word cloud diagrams, heat maps and other visualization methods to make data display more intuitive and efficient;
- This article not only quantifies the problem, but also combines reasonable qualitative analysis to make the main logic of the article more complete;
- The model has a variety of applications, and the results analysis is highly reliable. It has made a detailed and in-depth exploration of all aspects of data analysis involving online sales;
- The code in this article uses python language for programming, and its code has high generality. In addition, for the problem of natural language processing, this article proposes a relatively complete set of solution ideas, and the logical structure is relatively perfect. Therefore, With certain improvements and amendments, the model in this paper has higher generalization value.

9.2 Disadvantages

- Due to time constraints, the features and correlation indicators of the data cannot be fully mined, and the accuracy of the model needs to be further improved. This can also be used as the next research direction in this paper;
- Due to time constraints, more data could not be obtained, which limited our work accuracy to a certain extent.

10. A Letter to The Marketing Director

Dear Sir，

Thank you for inviting us as your Consultant. We have established a series of models to determine the relationship, metrics and parameters of reviews and stars to help the company succeed in three new online markets. The specific summary is as follows:

（1） From a platform management perspective

- Ensure the quality of listed products is reliable, and exquisite products can ensure stable praise, thereby driving more consumers to spend.
- According to market demand, add product functional design. For example, for hair_dryer, users pay great attention to its button functions to ensure sensitivity and controllability; for microwave, users pay great attention to its heating function; for pacifier, users pay more attention to its cute attributes and make targeted design.
- Strengthen the construction of the platform security mechanism to reduce the problem of evaluation without purchasing the products.These abnormal comments will not only mislead consumers on the one hand, but also not conducive to the analysis of the product data on the back end of the platform.
- Strengthen the management of the platform's fair mechanism, and prohibit competing businesses from taking measures to disrupt market order because of interest issues (such as sharp price reductions, malicious comments, etc.)

（2） From the perspective of consumption guidance

- Formulate product preferential policies to drive sales growth. In the analysis of the number of reviews and evaluation scores in this article, it is found that sales and reputation have a great correlation. Increasing sales will promote the improvement of reputation and enter a virtuous circle. Otherwise, enter Vicious circle.
- Actively guide consumers in high-star ratings, and develop measures to allow 3-star and 4-star users to give 5-star positive reviews (e.g., guarantee customer after-sales service; set positive rewards, etc.), because of the relevance of specific star ratings and reviews During the analysis, we found that high star ratings will greatly drive positive reviews, which is conducive to solidifying the reputation of the product market.
- When encouraging consumers to comment on text, use more positive and positive words such as "love", "enthusiasm", and "praise" (for example, arrange customer service for evaluation guidance; improve the platform evaluation system, and automatically select positive words, etc.). Because when analyzing the relationship between the sentiment keywords and comments in the text, we found that positive emotions have a higher probability of causing reviews of more than 3 stars, while negative emotions easily lead to low-star reviews. It is beneficial to improve product competitiveness and feasibility.

11. Reference

[1] Sharda R , Patil R B . Connectionist approach to time series prediction: an empirical test[J]. Journal of Intelligent Manufacturing, 1992, 3(5):317-323.

[2] Schroeder P , Dochow R , Günter Schmidt. Optimal solutions for the online time series search and one-way trading problem with interrelated prices and a profit function[J]. Computers & Industrial Engineering, 2018, 119.

[3] Esling P , Agon C . Time-Series Data Mining[J]. Acm Computing Surveys, 2012, 45(1):12.

[4] Weigend A S , Mangeas M , Srivastava A N . NONLINEAR GATED EXPERTS FOR TIME SERIES: DISCOVERING REGIMES AND AVOIDING OVERFITTING[J]. International Journal of Neural Systems, 1995, 06(04):373-399.

[5] Jaeger H . Observable Operator Models for Discrete Stochastic Time Series[J]. Neural Computation, 2000, 12(6):1371-1398.

[6] Chen M S , Han J , Yu P S . Data mining: an overview from a database perspective[J]. IEEE Transactions on Knowledge and Data Engineering, 1996, 8(6):866-883.

[7] Liu H , Motoda H . Feature Selection for Knowledge Discovery and Data Mining[J]. Springer International, 1998(4):xviii.

[8] Kanakaraj M , Guddeti R M R . NLP based sentiment analysis on Twitter data using ensemble classifiers[C]// 2015 3rd International Conference on Signal Processing, Communication and Networking (ICSCN). IEEE, 2015.

[9] Kanakaraj M , Guddeti R M R . Performance analysis of Ensemble methods on Twitter sentiment analysis using NLP techniques[C]// IEEE International Conference on Semantic Computing. IEEE Computer Society, 2015.

[10] Sindhwani V , Melville P . Document-Word Co-regularization for Semi-supervised Sentiment Analysis[C]// Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), December 15-19, 2008, Pisa, Italy. IEEE, 2008.

12. Appendix

```
import pandas as pd
import numpy as np
import os
from tqdm import tqdm
import seaborn as sns
from matplotlib import pyplot as plt
import pandas_profiling as pp
from wordcloud import WordCloud
get_ipython().run_line_magic('matplotlib', 'inline')
df_hair_dryer = pd.read_csv('../data/hair_dryer.csv')
df_microwave = pd.read_csv('../data/microwave.csv')
df_pacifier = pd.read_csv('../data/pacifier.csv')

from sklearn.feature_extraction.text import TfidfTransformer, TfidfVectorizer,
CountVectorizer

def get_TFIDF(documents, vector_size=512):
    vectorizer = TfidfVectorizer(token_pattern=r'(?u)\b\w+\b', max_features=vector_size)
    X = vectorizer.fit_transform(documents)
    return X.toarray()

tfidf_hair_dryer_review_body = get_TFIDF(df_hair_dryer.review_body)
tfidf_microwave_review_body = get_TFIDF(df_microwave.review_body)
tfidf_pacifier_review_body = get_TFIDF(df_pacifier.review_body)
from sklearn.manifold import TSNE
from sklearn.cluster import KMeans
kmeans = KMeans(n_clusters=10)
kmeans.fit(tfidf_hair_dryer_review_body[:1000])
print("inertia: {}".format(kmeans.inertia_))

tsne = TSNE(n_components=2)
decomposition_data = tsne.fit_transform(tfidf_hair_dryer_review_body[:1000])

x = []
y = []

for i in tqdm(decomposition_data):
    x.append(i[0])
    y.append(i[1])
xyc = [color[c] for c in kmeans.labels_]
fig = plt.figure(figsize=(12, 8))
ax = plt.axes()
```

```
plt.scatter(x, y, c=xyc)
plt.xticks(())
plt.yticks(())
# plt.show()
plt.savefig('./tfidf_v.png', aspect=1)

star_rating = df_hair_dryer.star_rating.loc[:1000].values - 1
color = sns.color_palette("hls",5)
xyc = [color[c] for c in star_rating]
fig = plt.figure(figsize=(12, 8))
ax = plt.axes()
plt.scatter(x, y, c=xyc)
plt.xticks(())
plt.yticks(())

file_name_list = ['hair_dryer', 'microwave', 'pacifier']
color_list = ['r','g','black']
df_list = [df_hair_dryer, df_microwave, df_pacifier]

fig1,axs=plt.subplots(1,3,figsize=(16,6))
column = 'RH_sentence_length'
title_name = ''
for i, ax in enumerate(axs):
    sns.distplot(df_list[i][column],ax=ax,color=color_list[i])
    ax.set_xlabel('Review headline length')
    ax.set_ylabel('Distribution')
    ax.set_title(f'Distribution of review headline length from {file_name_list[i]}')
fig1.show()
fig1.savefig('RH_sentence_length.png')

fig2,axs=plt.subplots(1,3,figsize=(16,6))
df_list = [df_hair_dryer, df_microwave, df_pacifier]
file_name_list = ['hair_dryer', 'microwave', 'pacifier']
column = 'RB_sentence_length'
title_name = ''
for i, ax in enumerate(axs):
    sns.distplot(df_list[i][column],ax=ax,color=color_list[i])
    ax.set_xlabel('Review body length')
    ax.set_ylabel('Distribution')
    ax.set_title(f'Distribution of review body length from {file_name_list[i]}')
fig2.show()
fig2.savefig('RB_sentence_length.png')

fig3,axs=plt.subplots(1,3,figsize=(16,6))
```

```
df_list = [df_hair_dryer, df_microwave, df_pacifier]
file_name_list = ['hair_dryer', 'microwave', 'pacifier']
column = 'PT_sentence_length'
title_name = ''
for i, ax in enumerate(axs):
    sns.distplot(df_list[i][column],ax=ax,color=color_list[i])
    ax.set_xlabel('product title length')
    ax.set_ylabel('Distribution')
    ax.set_title(f'Distribution of product title length from {file_name_list[i]}')
fig3.show()
fig3.savefig('PT_sentence_length.png')

fig4,axs=plt.subplots(1,3,figsize=(16,6))
df_list = [df_hair_dryer, df_microwave, df_pacifier]
file_name_list = ['hair_dryer', 'microwave', 'pacifier']
column = 'helpful_votes'
title_name = ''
for i, ax in enumerate(axs):
    sns.distplot(df_list[i][column],ax=ax,color=color_list[i])
    ax.set_xlabel('helpful votes')
    ax.set_ylabel('Distribution')
    ax.set_title(f'Distribution of helpful votes from {file_name_list[i]}')
fig4.show()
fig4.savefig('helpful_votes.png')
```