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

An MCM Paper

Summary

In the field of e-commerce, companies collect and analyze user evaluations to obtain user preferences and other information, and then improve the market competitiveness by improving some product attributes. We complete the analysis task through four steps: data preprocessing, feature extraction, model construction and model verification.

First, we eliminate redundant data and invalid records in the original data sets and transform the original data into feature vectors by combining with natural language processing (NLP) technology.

Next, we build two types of models: the basic model and time-based model. In the basic model, we use multiple linear regression model to get the internal relationship between user reviews and product sales, build a true likability model to obtain the true rating of the product considering star ratings and text reviews together, and employ frequent set mining algorithm, FPGrowth, to get the possibility of "bundled sales" between three types of products. In the time-based model, we use reputation decay model to obtain fluctuations in product reputation over a continuous time and determine whether the product is potentially successful of failing product, construct Markov model and the time-based K-Means algorithm to analyze whether the user has blind behavior of "herd comment", and employ Autoregressive Integrated Moving Average (ARIMA) model to predict product sales in the future based on historical sales data.

Further, we use the data sets provided by Sunshine company to verify our models and analyze results. The results show that our models fits well with the underlying rules in the data sets and is consistent with cognition in real life. For example, products with higher reputation tend to have higher sales.

Finally, based on analysis, we propose an online sales strategy and help Sunshine company to improve the attractiveness of your product by identifying potentially important design features.

Keywords: E-commerce, Natural language processing(NLP), time-based model.

Contents

1 Introduction

1.1 Problem Statement

With the rapid development of computer networks and modern logistics systems, modern e-commerce has flourished. Various e-commerce activities will generate a large number of transaction logs and process information data. Therefore, quickly and effectively obtaining the knowledge hidden in the information is very important in the current fierce commercial competition. The acquisition of knowledge requires data conversion, analysis, screening, sorting, organization, and utilization of this information, and finally looking for truly valuable data from it. Companies can use this data to determine when to participate and whether product design is successful.

As one of the earliest e-commerce companies on the Internet, Amazon provides a star rating mechanism to enable buyers to express their satisfaction with the product, and to express further opinions on the product based on textual information (comments). Other customers can submit ratings on whether these reviews are helpful for their product purchase decisions. We analyze the sales records of hair dryers, microwave ovens, and baby pacifiers for more than 10 years, perform mathematical analysis and modeling based on the processing of various attributes of the data, and make a series of practical suggestions to Sunshine company.

1.2 Current Situation

1.2.1 Related Work

(Naive Bayes Model.) Naive bayes model is a classification method based on Bayes' theorem and independent assumptions of feature conditions. It is one of the most widely used classification models with Decision Tree Model. The theoretical basis of Naive Bayes Model is Bayes formula, as displayed as following.

$$P(Y_k|X) = \frac{P(X|Y_k)P(Y_k)}{\sum_k P(X|Y = Y_k)P(Y_k)} \tag{1}$$

The naive bayesian model performs well on small-scale data and is suitable for incremental training, especially when the amount of data exceeds the memory. In addition, the model is less sensitive to missing data and the algorithm is simpler. The model can be incrementally trained in batches.

(Text Sentiment Analysis.) Text sentiment analysis refers to the process of analyzing, processing, and extracting subjective text with sentiment by using natural language processing and text mining techniques. At present, the research on text sentiment analysis covers many fields including natural language processing, text mining, information retrieval, information extraction, machine learning, and ontology. The sentiment analysis task can be divided into chapter level, sentence level, word or phrase level according to its analysis granularity. According to its processing text category, it can be divided into sentiment analysis based on product reviews and sentiment analysis based on news reviews. According to its research tasks types, it can be divided into sub-problems such as sentiment classification, sentiment retrieval and sentiment extraction. Now, there are many emotion recognition methods which can be

basically divided into two categories: one is based on temporal features, such as Hidden Markov Model; the other is based on statistical features, such as Artificial Neuro Network.

1.2.2 Our work

1. Clean up the data sets, such as deleting redundant columns such as market_place, product_category, and selectively removing some records, e.g., records with verified_purchase = N.
2. Use Textblob library and NLTK library in NLP technology to perform sentiment analysis on text-based reviews, and complete the conversion from text to feature vectors.
3. Construct a multiple linear regression model between user evaluation and product sales, and obtain the internal relationship between them.
4. Propose a true likability model based on user evaluation (i.e., star_rating, helpful_votes, total_votes, review_headline, and review_body), and verify the rationality of this model through product sales.
5. Build a reputation decay model by combining the true likability model and the exponential decay model to describe the reputation fluctuation of the product in continuous time.
6. Analyze the sales prospect (i.e., identify potentially successful or failing product) according to the change rate of product reputation, such as the possibility of a product becoming an explosive product.
7. Use Markov model and time-based K-Means algorithm to discover whether users have blind behavior of "herd comments".
8. Calculate the Pearson correlation coefficient between star rating and text-based review, and analyze the correlation between them.
9. Integrate all the data sets to obtain the user shopping lists, and employ FPGrowth algorithm to mine frequent itemsets to provide "combined product" marketing strategy.
10. Use ARIMA model to predict the future sales of products to guide the product storage plan, thereby avoiding the backlog of goods or the situation of oversupply.

In summary, we employ a total of five models, that is, multiple linear regression model, true likability model, reputation decay model, Markov model and ARIMA model, two machine learning algorithms (i.e., K-Means and FPGrowth), and NLP technology to solve all the requests of Sunshine company, and give our business recommendations.

2 Model Development

2.1 Assumption

- Assume that records in the original data sets are real, especially the comments, which cannot be maliciously tampered with due to human factors.
- In the time-based models, we assume that each month is 30 days for statistical and charting purposes.
- Assume that there is no irony or special use of the words in the comments, which leads to the program's misjudgment.
- When consumers visit e-commerce websites, they have different trust degrees for star rating and text comment, and we assume that the weights of star rating and comment are 30% and 70% respectively.

2.2 Notations

The primary notations used in this paper are listed in Table 1.

Table 1: Notations

Symbol	Definition
Hv	the value of helpful_votes field
Tv	the value of total_votes field
Pi	product id
Pp	product_parent id
Sr	the value of star rating
Spc	score per comments
Tsc	total score of comments
X_1, X_2	the number of positive comments
X_3	the number of neutral comments
X_4, X_5	the number of negative comments
$TLF(i)$	true likability of product(i)
Pos	score of positive comment
Neg	Score of negative comment

2.3 Multiple Linear Regression Model

Linear regression is a statistical analysis method that uses regression analysis in mathematical statistics to determine the interdependent quantitative relationship between two or more variables. Although its form is simple, it contains some important relationships between the data. Therefore, linear regression model is widely used. The expression of the linear regression model is shown as follows.

$$hw(x) = w_1x_1 + w_2x_2 + ... + w_dx_d + b \tag{2}$$

among them, (w) is the weight, (b) is the bias term, (x) is the independent variable (i.e., the characteristic), and (hw(x)) is the dependent variable (i.e., the target value). According to the number of independent variables involved, the linear model can be divided into unitary linear model (i.e., only one independent variable is involved) and multivariate linear model (i.e., more than two independent variables are involved).

In order to describe the error between the actual value and the predicted value, the loss function is introduced, and its definition is as follows.

$$J(w) = (hw(x_1) - y_1)^2 + ... + (hw(x_m) - y_m)^2 = \sum_{i=1}^{m} (hw(x_i) - y_i)^2 \tag{3}$$

among them, y_i is the true value of the i_{th} training sample, and $hw(x_i)$ is the predicted value of the eigenvalue combination of the i_{th} training sample. The smaller the loss function, the better the fitting effect. At present, there are two methods to obtain the regression coefficient (i.e., W) in the model to minimize the loss: normal equation and gradient descent. The normal equation directly obtains the optimal weight according to the eigenvalue matrix and the target value matrix, but when there are too many features, the solution speed is too slow, and therefore it is not a universal method. The gradient descent is a direct method to find the minimum value of the loss function, that is, to update the weight combination along the direction of the fastest decline of the loss function (i.e., the gradient) until the minimum value of the loss function is reached.

In our analysis, we first obtain 5-dimension feature vectors based on the attributes of star ratings, reviews, and helpfulness ratings of each product, that is, the number of positive reviews, the number of middle reviewers, the number of bad reviewers, helpful_votes value and total_votes value, and then determine the relationship between feature vectors and product sales using the multiple linear model. In the implementation, we use stochastic gradient descent (SGD) to get weights of this model. This model can help Sunshine company infer product sales based on user evaluation, so as to guide product supply planning, such as producing the right amount of products to avoid product backlogs or supply shortage.

2.4 True Likability Model

Combining the star rating and text review on products, we define an data measurement that describes the product, called true likability, which is expressed as follows.

$$TLF(i) = \begin{cases} \frac{\sum_{all} Sr}{5} \times 0.3 + [Spc \times (Hv{+}1) + (\frac{Tsc+1}{2}) \cdot (Tv - Hv)] \times 0.7 (\text{Spc} < 0) \\ \frac{\sum_{all} Sr}{5} \times 0.3 + [Spc \times (Hv{+}1) + (\frac{Tsc-1}{2}) \cdot (Tv - Hv)] \times 0.7 (\text{Spc} \geq 0) \end{cases} \tag{4}$$

where i is product_parent id, $TLF(i)$ is the true likability of the product, Sr is the star rating score, Spc is the score of the text review, Hv is the number of users who think the review is useful, and Tv is all the users who pay attention to the review.

As for Sr, we think that the user star rating of 4 or 5 is the favorable rating, which will improve the overall likability, the star rating of 1 or 2 is the poor rating, which will reduce the overall likability, and the star rating of 3 is the medium rating, which will not affect the likability. Therefore, the mapping relationship between Sr and user rating star is established as follows. note that we divide the sum of stars by 5 for the

Table 2: The mapping relationship between Sr and Star Rating

Star Rating	Sr
1	-5
2	-4
3	0
4	4
5	5

purpose of normalization.

For Spc, we think that the text comments reflect the user preference and aversion and other emotional tendencies to the product, and therefore we use Textblob in NLP technology to carry out emotional analysis on the text comments to obtain Spc. When Spc is less than 0, it means the comment is negative; when the score is greater than 0, it means the comment is positive. Meanwhile, the closer Spc is to 1, the stronger the favorable impression is, the closer it is to -1, the stronger the aversion is.

Meanwhile, we also considered the influence of helpful_votes and total_votes fields. We think that other users pay more attention to text comment than to star rating (that is, text review better reflects user psychology). Therefore we use Hv and Tv to process text comment scores. Specifically, since the user has the same evaluation on the product as other Hv users, $(Hv + 1) \times Spc$ is obtained. For $Tv - Hv$ users who do not agree with the current text comment, their comments are the opposite of the current comment, and we compromise the text reviews of these users.

The reason for not averaging the two parts of the model is: for example, product A was evaluated by 100 people and all of them scored 5 points, while product B was only evaluated by 1 people and scored 5. If the average is calculated, the two products will have the same score of true likability, but obviously product A has a higher likability.

Finally, we think that when consumers visit e-commerce websites, they have different trust degrees for stars and text comments. Here, we divide the weight of stars and comments into 30% and 70% respectively. In addition, in the specific implementation of the model, we divide users into two types, that is, vine user and non-vine user. We calculate the true likability of the two types of users separately, and then calculate the weighted average. Note that since vine users have high credibility, a higher weight should be set.

2.5 Reputation Decay Model

A user comment and star rating will have a certain impact on the "heat" (or reputation) of the product, but this effect will slowly decay over time. For example, a positive review in the last week and a positive review two years ago will obviously make a difference to current users. We abstract the impact of user reviews on product reputation as a natural cooling process over time. Inspired by Newton's cooling law, we think that for each user evaluation, the decay rate of reputation is directly proportional to the current reputation. The mathematical formula is:

$$\frac{dN}{dt} = -\alpha N \tag{5}$$

where $\alpha(\alpha > 0)$ is called the decay constant. We can get the result by solving the differential equation:

$$N(t) = N_o e^{-\alpha t} \tag{6}$$

among them, $N(t)$ is the value of N at time t, and $N_0 = N(0)$ is the initial value of N at time 0, which is also the maximum value when $0 \leq t$. For example, if the initial reputational impact of a comment is N_0, it can be concluded from the above formula that its reputational impact declines exponentially over time. Therefore, this decay model is also called exponential decay model.

Through the extended exponential decay model, we further propose a reputation decay model with multiple user comments, that is, the cumulative reputation value or the overall reputation value of product can be obtained by accumulating the reputation value after each decay. The model is as follows.

$$N(t) = \sum_{k=1}^{M} N_i \cdot e^{-\alpha(t-t_{\mathrm{i}})} \tag{7}$$

among them,$N(t)$ is the overall reputation of the product in month t, Ni is the initial reputation impact of the comment i on the product, t_i is the release time of the comment i, and M is the number of comments before time t. In the reputation decay model, we calculate the initial reputation value of the comment i by using the true likability model. α shows the decay rate of reputation. We assume that the "average life" of a comment is 1 month, that is, its reputation value is $1/e$ of the original reputation value after a month, and therefore we set the decay rate a to 1.

In summary, the reputation decay model uses time and user evaluation to determine the product's cumulative reputation. In the analysis, we use the reputation decay model to obtain the reputation value of three types of products for 12 consecutive months, showing the changes in product reputation over a continuous time, and analyzing the relationship between reputation and product sales.

2.6 Markov Model

Markov Model is a statistical Model, widely used in speech recognition, automatic part-of-speech tagging, and other applications. After a long period of development, especially in the successful application of speech recognition, it becomes a universal statistical tool.

The original Markov model is based on Markov chains. Markov chain refers to the random process of discrete events with Markov property in mathematics. At each step of the Markov chain, the system can either go from one state to another or remain in its current state, depending on the probability distribution. A change in state is called a transition, and the probability associated with different state changes is called a transition probability. The sum of all the states can be called the state space. As shown in the following figure, circles represent different states, the starting point of arrows represents the source of state transitions, the circle pointed by arrows represents the purpose of state transitions, and the weight on each arrow represents the probability of state transitions. Markov model can summarize the law of change based on historical data, so as to use the form of probability transfer graph to infer or fit some facts.

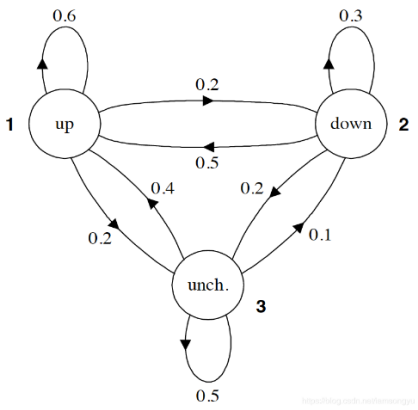

Figure 1: maerkefu

2.7 ARIMA Model

The full name of the ARIMA model is Autoregressive Integrated Moving Average Model, also known as $ARIMA(p, d, q)$. It is the most common statistic model used for time series prediction, where AR represents the autoregressive process, and p is the autoregressive term. MA is the moving average process, q is the number of moving average terms, and d is the number of difference made when the time series become stationary.

In the analysis, we use the ARIMA model to predict the future sales volume of the product in monthly units, so as to guide Sunshine company product storage plan (making the existing products close to the predicted sales volume), so as to avoid problems such as backlog of products. Note that when using the ARIMA model to predict time series, it is necessary to require the time series to be stationary or to be stable through differencing, and therefore we first convert the original time series to a stable one.

2.8 K-Means and FPGrowth Algorithm

K-Means K-Means clustering algorithm is an iterative cluster analysis algorithm. Its steps are: pre-divide the data into K groups, and randomly select K objects as the initial cluster center; calculate the distance between each object and each cluster center, and assign each object to the nearest cluster center. The cluster centers and the objects assigned to them represent a cluster. For each sample assigned, the clustering center of the cluster is recalculated based on the existing objects in the cluster. This process is repeated until a certain termination condition is met. The termination condition may be that no (or minimum number) objects are reassigned to different clusters, no (or minimum number) cluster centers change again, and the squared error and local minimum.

In the analysis, we use time based K-Means to cluster the user reviews in each month according to the time sequence, so as to obtain the proportion between the good and bad reviews in a continuous time. If the ratio between good and bad reviews of the product is roughly stable over a continuous period of time, it can be inferred that reviews are almost all objective reviews. If the ratio between good and bad reviews of the product fluctuates obviously, it is speculated that there may be a phenomenon that users follow the comment trend, for example, seeing that most users comment badly,

and also give bad comments.

FPGrowth Frequent itemsets mining algorithm is used to mine the item sets that often appear together (called frequent itemsets). By mining these frequent itemsets, when one item of the frequent itemsets appears in a transaction, other items of the frequent itemsets can be taken as recommendations, such as the story of beer and diapers in the classic shopping basket analysis. Beer and diapers often appear together in a user's shopping basket. By digging out frequent itemsets such as beer and diapers, a user can recommend a diaper when he buys beer, so that the user is more likely to buy it, thereby achieving the purpose of combined marketing.

In our analysis, we apply the FPGrowth algorithm to verify the possibility of "combination marketing" between hair dryer, pacifier and microwave (i.e., users are more likely to buy another product when they buy one product). Specifically, we first remove invalid data from three files, then integrate the shopping list of all users, and finally use FPGrowth algorithm to find frequent sets with support greater than the set value. Such frequent sets can guide the marketing strategy of Sunshine company's products. For example, if product A and product B are A frequent set, the marketing strategy can be that buying A product makes 20% off on B product.

3 Model Application and Problems Solution

3.1 Request 1: Relationship between Three Factors and Product Sales

3.1.1 Problem Solution

We use the multiple linear regression model described in section 2.3 to depict the relationship between sales volume and user reviews.We abstract user comments into a 5-dimension eigenvector, that is, the number of good comments, middle comments, bad comments, helpful_votes and total_votes.Among them, we classify those with star rating of 4 or 5 as good comment, 3 as middle comment, 1 or 2 as bad comment. According to this rule, we can get the records of 5-dimension eigenvector set and target value of each type of product. In addition, we use the least square method to obtain the linear regression coefficient. According to the number of training samples used to update model parameters after each iteration, there are two gradient descent methods: Batch gradient descent (BGD), in which all training samples are used in each iteration; Stochastic gradient descent (SGD) every iteration USES a training sample, which is randomly selected. A training set of batch gradient descent method can only produce one result, while SGD will produce different results every time it runs, which makes SGD easy to jump out of the "local optimal solution". Therefore,when there are more training samples, the stochastic gradient descent method can find the optimal parameters faster than the batch gradient descent method. It is also worth noting that SGD may not ?nd the minimum because only one training sample is used to update the weight, but its approximation is usually close enough to the minimum. For the specific implementation, we used sklearn in machine learning library to cross-train the model, and to prevent overfitting, we also conducted regularization of the data.

3.1.2 Sensitivity Analysis

Through the multiple linear regression model, we obtained the comparison figure between the fitting sales volume and the actual sales volume of the hair dryer, as shown in the following figure:

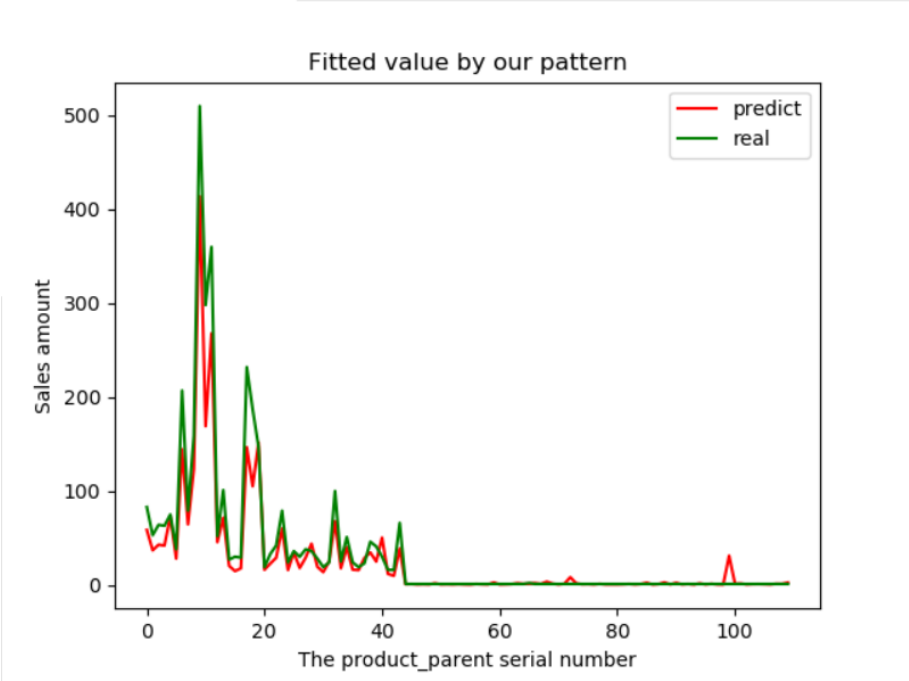

Figure 2: Fitted hair_dryer's value by our model

This figure shows the difference in sales between different product_parent, We can see that the effect of fitting hair dryer's sales volume through multiple linear regression model is great. Goodness of Fit R^2 equals 0.9848932283, the trend of forecast and actual sales is very similar, the mathematical model is:

$$hw(x) = 0.38164x_1 + 0.25932x_2 + 0.21767x_3 + 0.09328x_4 + 0.08307x_5 - 0.00012 \quad (8)$$

We conclude that the relationship between star ratings, reviews, helpfulness ratings and sales volume conforms to the multiple linear regression model.

Due to the space limitation, the model fitting diagram and mathematical model of the pacifier and microwave oven are placed in Appendix A.

3.2 Request 2.a: Identify Measures based on Ratings and Reviews

3.2.1 Problem Solution

We identify a data measure based on ratings and reviews, called likability. In order to calculate the product likability, we build the likability model for the vine users and the no-vine users respectively (the details are described in 2.4). Then, through the method of weighted average to get the overall likability of the product.Therefore, the likability model of the product is shown as follows:

$$TLF(i) = 0.7 \times TLF\_Y(i) + 0.3 \times TLF\_N(i) \quad (9)$$

$TLF(i)$ is the overall likability of product parent i, $TLF_Y(i)$is the overall likability of vine, $TLF\_N(i)$ is the overall likability of no-vine. Since vine has a good reputation, we believe that their reviews of the product are more objective. We set a higher weight for vine ,which is set to 0.7. Since we need to process text comments in building the likability model, we call the natural language processing library TextBlob to perform sentiment analysis on the text built with "review_headline" and "review_body."

3.2.2 Sensitivity Analysis

Through the true likability model, We analyzed the top 10 hair dryers, microwave ovens and pacifiers respectively. The results obtained from the pacifier are shown in the following figure:

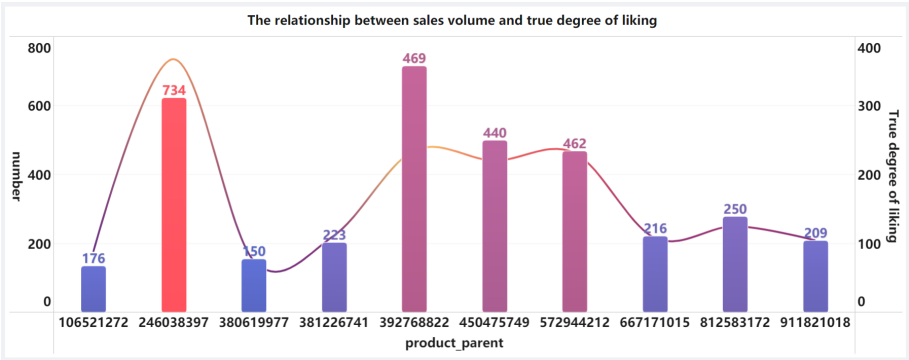

Figure 3: The relationship between pacifier sales and true liking

According to the figure, the change of sales volume is roughly similar to the trend of the change of the product's true favorable impression, which is positively correlated. Therefore, we believe that the star rating and comments will affect the product's likability, thus affecting the product's sales volume.

Other model figures are added to the Appendix A.

3.3 Request 2.b: Identify the Trend of Productąŕs Reputation

3.3.1 Problem Solution

To identify the trend of product reputation, we employ Reputation decay model to obtain the reputation of the product for a continuous time. For a detailed explanation of this model, see section 2.5.

We first select the top 10 products from the each type of products (hair dryer, baby bottle, and microwave oven), and then count the records of each product by month, and built reputation decay models. Similarly, in the feature extraction process, we use TextBlob to perform sentiment analysis on the text constructed by "review_headline" and "review_body" fields to obtain the feature value of the text review.

3.3.2 Sensitivity Analysis

In this problem, we respectively select products with best sales of hair dryer, microwave oven and pacifier. Then calculating the reputation change of each month

through the reputation decay model from September 2014 to August 2015 . The results are shown in the following figure:

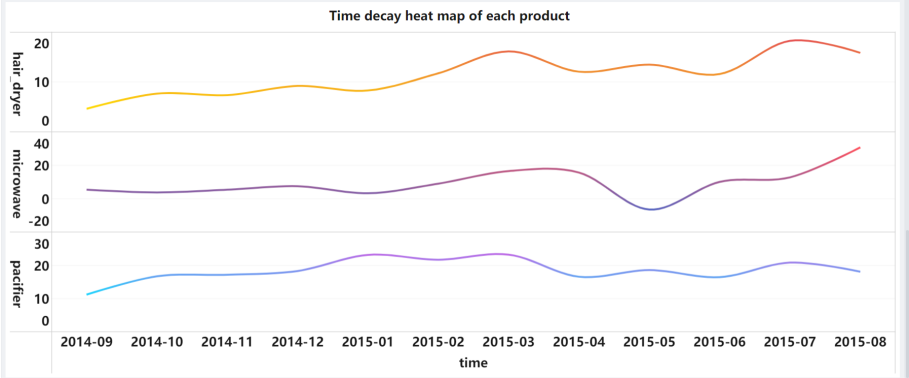

Figure 4: The relationship between pacifier sales and true liking

We found that for the same product, reputation fluctuates rises and falls over time. For example, the reputation of the microwave oven hit bottom in May 2015, sales of the product 423421857(microwave) also dropped in the last month. Looking at the data in mircowave.tsv, we also found that this product's the ratings and reviews were bad during this period. According to the comments, the problem of product quality caused the reputation drop during this period. After May 2015, reputation recovered and sales picked up slightly.

3.4 Request 2.c: Indicate Potentially Successful or Failing Products.

3.4.1 Problem Solution

We define the product which has a continuously growing reputation as the successful product, and the product with a continuously decreasing reputation as failing product. Therefore, we can discover potentially successful of failing production via deriving Reputation decay modelčňthat is, identifying the potential of product sales based on the rate of change in reputation. Its calculation model is shown below.

$$reputation\_decay\_rate = \frac{\partial N(t)}{\partial t} \tag{10}$$

We chose the three best selling products among three data sets, construct their Reputation decay models, and then calculate the product's reputation change rate per month according to formula 2.

3.4.2 Sensitivity Analysis

When the function value is above 0, it means that the product's reputation is positive in this time period and there is a high probability of success after purchasing goods at this time.

When the function value is under 0, it means that the product's reputation is negative in this time period and there is a high probability of failure after purchasing goods at this time.

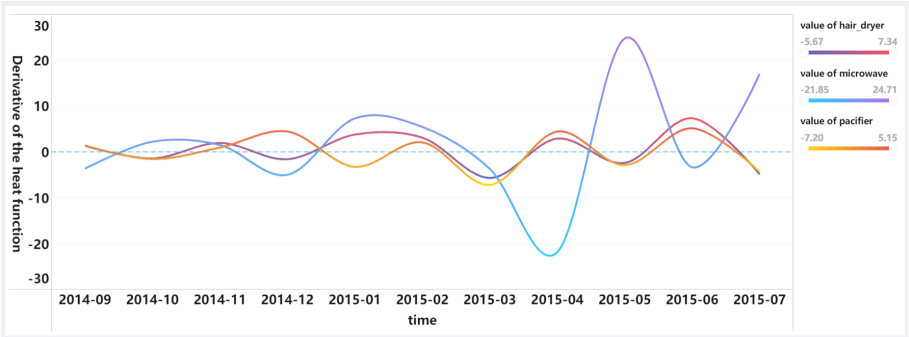

Figure 5: Analysis of reputation decay over time

When the function value is 0, we need to reference the reputation of the previous period.When the previous value is negative number, this means the reputation of the product is rising, at this point may be successful after purchase. When the previous value is positive number, this means the reputation of the product is falling, at this point may fail after purchase.

3.5 Request 2.d: Analyze Relationships among Reviews

3.5.1 Problem Solution

We use Markov model and K-Means algorithm based on time series to discover the relationship between user reviews. For only star rating, we try to use the first-order Markov model to analyze the relationship between the current user's star rating and the next user star rating. In this model, the state space if from 1star to 5 stars, and the state transition probability matrix is obtained by counting data set. We select three products with the best sales among the three product categories, and analyze the relationship between userąŕs star ratings.

Considering text based reviews and stars together, we employ K-means algorithm based on time series to discover the ratio change between positive and negative reviews over time. Since the traditional K-Means algorithm is easily affected by the selection of the initial clustering center point, we employ Bisecting K-Means to achieve clustering. Specifically, Bisecting K-Means clusters user's reviews in each month according to the time sequence, so as to obtain the ratio between the good and bad reviews in a continuous period of time. According to the stability of this ratio we can analyze the relationship between userąŕs reviews.

3.5.2 Sensitivity Analysis

The state transition matrix is as following Figure 6.

Results show that there is no "follow-up comment" phenomenon among the three types of products based on star rating analysis.

Through the clustering analysis of 12 months' pacifier data by K-Means algorithm, we obtained the following Figure 7.

We found that there was little change between positive and negative reviews in the past 12 months, which was ordinary fluctuations. Therefore, we came to the conclusion that specific star ratings do not incite more reviews.

	1 Star	2 Star	3 Star	4 Star	5 Star
1 Star	0.102	0.041	0.143	0.408	0.306
2 Star	0	0.167	0.111	0.278	0.444
3 Star	0.097	0.024	0.098	0.293	0.488
4 Star	0.154	0.018	0.09	0.27	0.468
5 Star	0.133	0.057	0.103	0.253	0.454

(a) Microwave

	1 Star	2 Star	3 Star	4 Star	5 Star
1 Star	0	0	0.133	0.1	0.767
2 Star	0.069	0	0.034	0.207	0.69
3 Star	0	0.043	0.065	0.087	0.805
4 Star	0.02	0.07	0.03	0.15	0.73
5 Star	0.041	0.032	0.056	0.117	0.754

(b) Pacifier

	1 Star	2 Star	3 Star	4 Star	5 Star
1 Star	0.028	0.028	0.25	0.194	0.5
2 Star	0	0.138	0.069	0.172	0.621
3 Star	0.07	0.035	0.105	0.194	0.596
4 Star	0.055	0.045	0.055	0.145	0.7
5 Star	0.071	0.048	0.096	0.2	0.585

(c) Hair dryer

Figure 6: Markov state transfer matrix

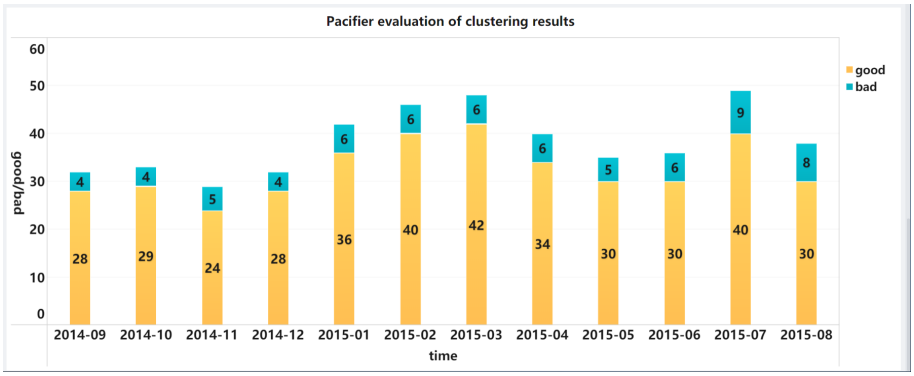

Figure 7: Pacifier evaluation of clustering results

Other model figures are added to the Appendix A.

3.6 Request 2.e: Analyze Relationships between Reviews and Ratings

3.6.1 Problem solution

We obtain the correlation between text reviews and star ratings by calculating the Pearson correlation coefficient. The formula for the Pearson correlation coefficient is as follows:

$$r(X,Y) = \frac{Cov(X,Y)}{\sqrt{Var[X]Var[Y]}} \tag{11}$$

Where Cov(X,Y) is covariance of star vector and text review vector, Var[X] is variance of star vector, and Var[Y] is variance of text review vector.

3.6.2 Sensitivity Analysis

We did the correlation analysis between text-based reviews and rating levels by FPGrowth algorithm, we obtained the following figure:

We collected 1095 effective records of microwave ovens, and gave the different distribution between specific quality descriptors of text-based reviews and rating levels. Finally, we concluded that the Pearson correlation coefficient of microwave ovens is 0.5449, belonging moderately relevant.

The Pearson correlation coefficient of hair dryer is 0.5081, belonging moderately relevant. The Pearson correlation coefficient of pacifier is 0.4718,belonging moderately

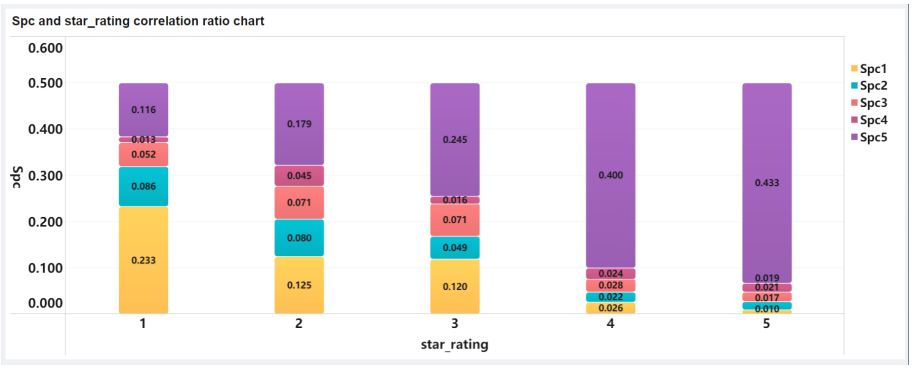

Figure 8: The microwave's correlation between reviews and rating levels

relevant.The figures for them are added to the Appendix A.

4 Strengths and Weaknesses

4.1 Strengths

Our models take into account user evaluation, evaluation of user evaluation, time and other factors, and process all three data sets. Based on the processing results and analysis, we list the advantages of models as follows.

- **Accuracy:** The accuracy of our model is very high, for example, the multiple linear regression model can accurately predict the product sales according to the comments of users, and the product likability obtained by the true likability model also has a significant positive correlation with product sales.
- **Universality:** Our model has an impressive processing effect on all of three data sets, and therefore we can consider our model to be equally applicable to other product types.
- **Flexibility:** When we test all time based models (such as reputation decay model), we use month as the unit, but this unit can be modified flexibly according to specific requirements, such as day, quarter, year, etc.

4.2 Weakness

- **Inadequate processing of text comments:** Actually, we only use NLP technology to conduct emotional analysis of users' text comments, that is, to obtain the user's satisfaction degree reflected in the text comments. In addition, the text review may include more information, such as the user's views on the attributes of products such as delivery, size, customer service, etc., from which it can further summarize which attributes of products most affect the user rating.
- **Inadequate products joint research:** Based on the three data sets, we integrate the user's purchase list, and find whether there is a possibility of bundling between products based on frequent set mining (that is, one product has a higher probability to buy when another product is bought), and no further research has been done on the link links between products.

4.3 Future Work

First, add a thesaurus training mechanism based on our model, and refine the classification of natural language sentiment analysis, such as distinguishing the user's views on various attributes of goods in text reviews (e.g., logistics, size, customer service, etc.), and further summarize which attributes of the product affect user evaluation.

Second, consider a deeper analysis of whether the same user has a tendency to buy in combination for different product types, and clustering analysis using a Gaussian mixture model (GMM) based on the existing model.

Finally, in fact, the change in product heat is not just a natural cooling process, because there will be explosions in the product. If we apply the logistic function to our model, and let the result of this function be the decay rate ę, then this ę will increase and then decrease over time, which is more in line with the change in the popularity of explosive products.

5 Conclusion

From the multiple linear regression model we fitted, there was a strong quantitative relationship between user evaluation and sales. Under the premise of constant product quality, merchants can invite customers who are invited to become Amazon Vine Voices to advertise , thereby increasing product reputation and driving product sales. From our reputation decay model, product reputation is affected by sales, reviews, likability, and other factors, and the fluctuation of the reputation change rate was related to future sales. When only star rating was considered, Markov model found that there was no conformity phenomenon in star rating evaluation of all products. However, when combining text reviews and star ratings, the time-based K-Means algorithm found that there was a certain degree of herd phenomenon in user reviews of hair dryers. In addition, we also found that the correlation between star ratings and text reviews is around 0.5, which is moderately relevant. Users who gave 1 star had the highest probability of writing bad reviews. Similarly, users who gave 5 stars had the highest probability of writing positive reviews. However, there were still some extreme manifestations, for example, 1 star rating mapped 5 stars reviews, and 5 star rating mapped 1 star reviews.

Memorandum

To: The Marketing Director of Sunshine Company
From: Team ######
Date: March 9st, 2020
Subject: Analysis and recommendations from Team ######

I write to you on behalf of the MCM team #####. We are extremely concerned about the current development situation of e-commerce industry. Investigation shows that e-commerce manufacturers can obtain user preferences and other information based on user comments, so as to improve product attributes and improve economic benefits.Through our analysis, we hope to help Sunshine company better understand the relationship between products and users, and design a reasonable marketing plan.

We use multiple linear regression models based on user evaluations to get predicted product sales. Based on this, you can adjust the product inventory plan in time to ensure the reasonable use of inventory resources and improve market efficiency. Specifically, when the forecasted sales volume is higher than the current sales volume of the product, you should replenish the product in a timely manner to avoid causing a shortage. When the predicted sales volume is lower than the current sales volume of the product, you should stop replenishing the product to avoid the backlog of the product.

We define a informative data measure based on ratings and reviews, called true likabilityčňwhich can help you to track the most popular product. The true likability model comprehensively considers the user type, star rating, text evaluation, helpful_votes, and total_votes fields, and uses a weighted average calculation to obtain a comprehensive likability measurement for each product. According to our analysis above, there is a positive correlation between the product's true likability and product sales, so grasping the product's true likability is equivalent to grasping the product sales.

We use a time-based reputation decay model to determine changes in product reputation. Specifically, we build the reputation decay model based on an exponential decay model and a true likability model. Reputation decay models can help you gain visibility into product reputation fluctuations over time. When the product reputation is relatively stable for a period of time, it indicates that the product sales are in a normal state. When the reputation of the product fluctuates greatly, it may be that some users maliciously smear the product, which reduces the reputation. When this happens, we recommend that you deal with these malicious comments in a timely manner.

In addition, we differentiate the reputation decay model to obtain the change rate of product reputation. When the change rate is positive, it means that the product's reputation will continue to increase, especially when the change rate exceeds a positive value, it means that the product's reputation has suddenly increased, and that the product may become an explosive product. When the change rate is negative, it indicates that the product reputation is in the process of decline. At this time, you should increase the promotion of the product or use some promotional methods to make the product attract the attention of consumers.

We use Markov model and time-based K-Means clustering algorithm to analyze whether there is a blind ąřherd commentąś behavior among users. When only star rating is considered, the result of Markov model is that there is no ąřherd commentąś

behavior among users in the three data sets. When considering star ratings and text reviews, the time-based K-Means algorithm yields that there is a certain degree of herd mentality among users' reviews of hair dryers. At this time, it is recommended that you adopt some encouragement mechanisms (e.g., cashback) to make users based on their objective use case to comment.

We calculated the Pearson correlation coefficient between text comments and stars. The correlation coefficients were 0.4718, 0.5449 and 0.5081, respectively, for the three data sets of pacifier, microwave oven and hair dryer. Therefore, it is concluded that there is a moderate degree of correlation between text reviews and star ratings.

We comprehensively process the three data sets, generate shopping records for all users, and applied the frequent set mining algorithm FPGrowth in an attempt to discover the possibility of "combination sales". However, based on the three existing data sets, we obtain a total of 26,033 users' shopping lists, of which only 947 users (about 3.6%) purchase more than two products, including products of the same type. Therefore, it is concluded that according to the available data, we do not consider it necessary to "bundle" between the products, that is, there is no relationship between the three products like beer and diapers.

We use the ARIMA model to predict monthly product sales. Through the processing of historical data, we use the ARIMA model to obtain the product sales situation in the future. Based on this, you can more reasonably arrange the production plan, so that the sales volume and production volume are balanced.

The above is all our analysis, we hope this letter help you.

Sincerely yours

MCM 2020 Team #####

References

[1] Tan peng, Luo shunlian, Sun xiaosong, Wang hui, Liang xiaohan. Research on prediction model of topic heat based on wavelet neural network [J]. *Modern information technology*, 25 May 2018, 2 (5) : 74-78.

[2] Yang erhong, Zhang guoqing, Zhang yongkui.A method of Chinese word sense disambiguation based on the co-occurrence frequency of yiyuan. *Computer research and development*, 2001, 38 (7) : 834-837.

[3] Yarowsky,D.Word-Sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora.*In Proceedings.COLING-92.Nantes*, 1992:454-460.

[4] Shao hongxiang. Application and improvement of linear regression method in data mining [J]. *Statistics & decision*, 2012, (14) : 76-80.

[5] Li peizhe. Grey multiple linear regression model and its application [J]. *Statistics and decision*,2012, (24) : 89-91.

[6] Cohen,J,Cohen P, West,S.G,& Aiken,L.S.Applied multiple regression/correlation analysis for the behavioral sciences. *Hillsdale*,NJ: Lawrence Erlbaum Associates. 2003.

[7] Liu zhihua, liu ruijin. Cooling law of Newton cooling law [J]. *Journal of shandong university of science and technology (natural science edition) (6)* : 23-27.

[8] Ding jiangwei, liu ting, lu zhimao, et al. Comparative study of hidden markov model and bayesian model word sense disambiguation [C] *computational linguistics*, 2003:3-4.

[9] Chen xiaoyan. Application of machine learning algorithm in data mining [J]. *Modern electronic technology*, 2015(20) : 19-22.

[10] Mo xuefeng. Application of machine learning algorithm in data mining [J]. *Science and education (21)* :175-178.

[11] Jiang chengyu, Sun deshan. Forecast of beverage sales based on ARIMA model [J]. *Modern business*,2009 (30) : 145-145

Appendices

Appendix A First appendix

Time series analysis by ARIMA

We first sort out the monthly sales volume of the three products, and then employ time series prediction model, ARIMA, to predict product sales in the future. Results are shown as Figure 9, Figure 10, Figure 11. According to this, the merchant can more reasonably arrange the production plan to achieve a balance between sales and production.

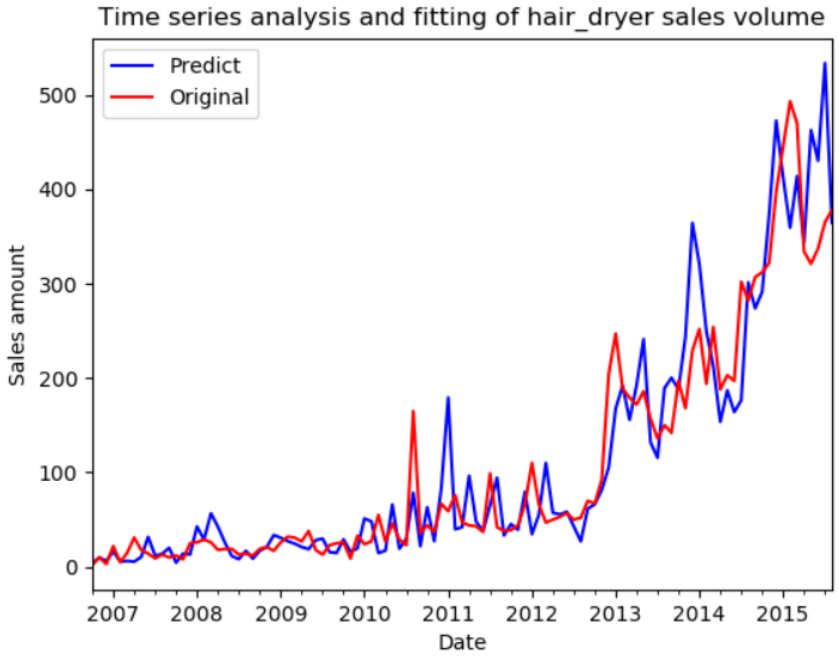

Figure 9: Time series analysis of hair dryer

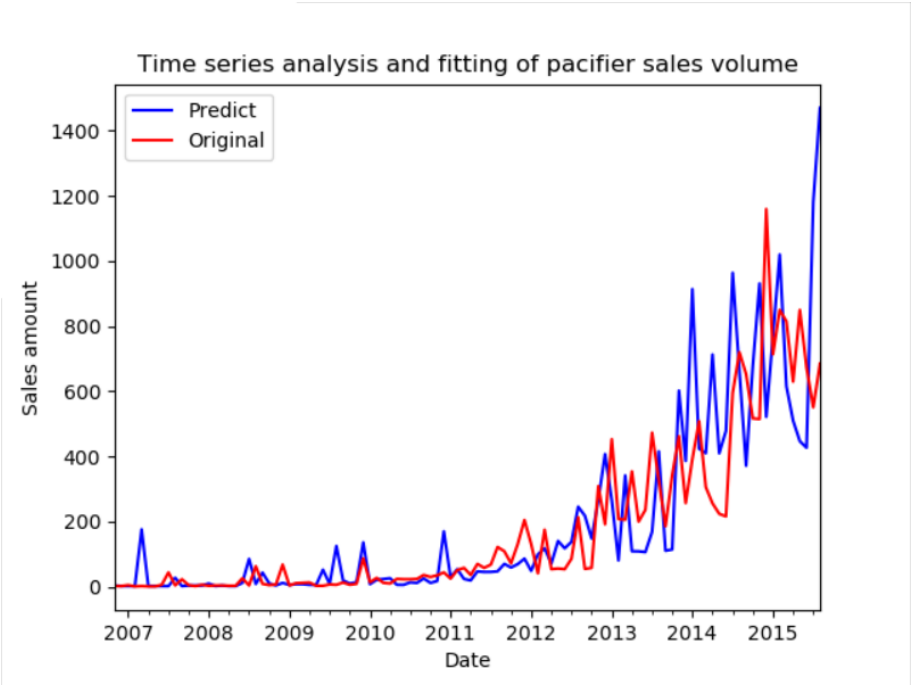

Figure 10: Time series analysis of pacifier

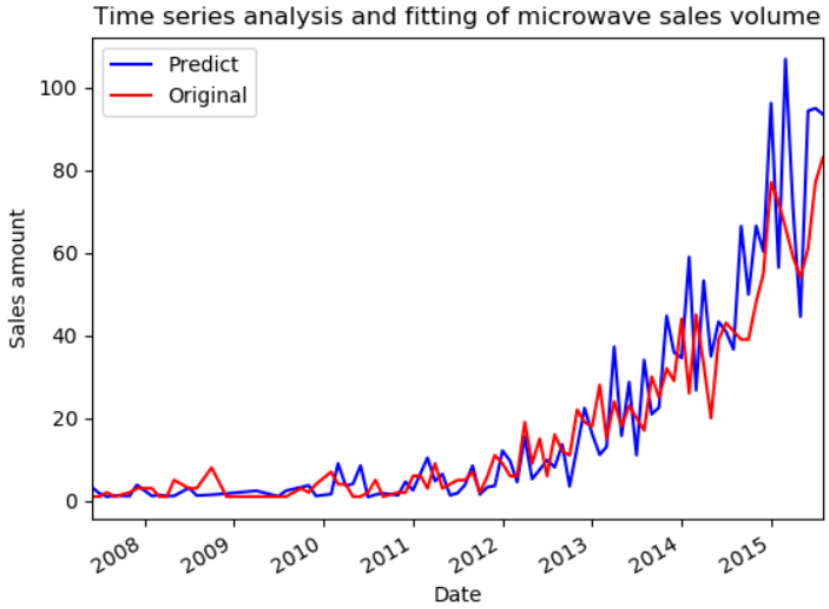

Figure 11: Time series analysis of microwave

Problem a - rest of the pictures and formulas

The mathematical model of pacifier:

$$hw(x) = 0.77453x_1 + 0.0668x_2 + 0.06521x_3 + 0.10001x_4 + 0.00468x_5 - 0.00013 \quad (12)$$

The mathematical model of microwave:

$$hw(x) = 0.24484x_1 + 0.11709x_2 + 0.20264x_3 + 0.22467x_4 + 0.22465x_5 + 0.0099 \quad (13)$$

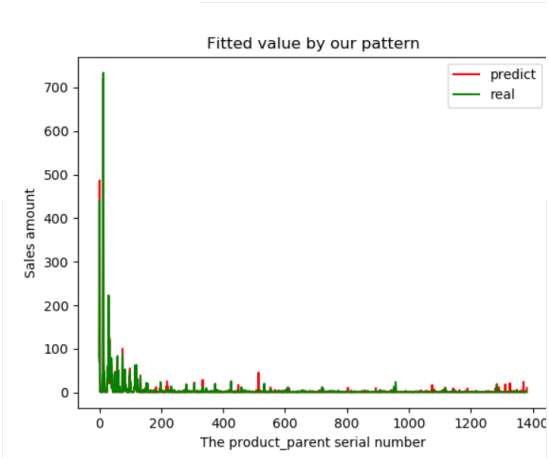

Figure 12: Fitted pacifier's value by our model

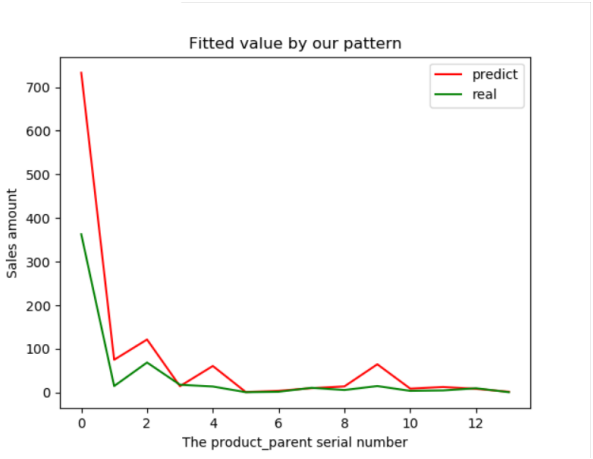

Figure 13: Fitted pacifier's value by our model

Problem d - rest of the pictures and formulas

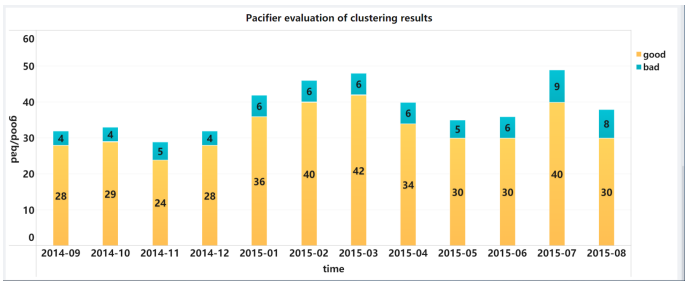

Figure 14: Hair dryer evaluation of clustering results

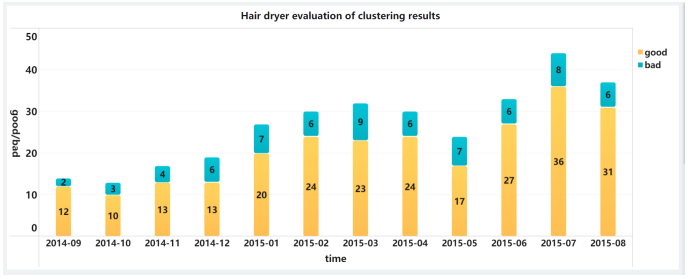

Figure 15: Microwave evaluation of clustering results

Appendix B Second appendix

Here is part of python programs we used to get the relationship between three factors and product sales

Input python programs:trian_hair.py

```
from sklearn.linear_model import SGDRegressor
from sklearn.model_selection import cross_val_score, train_test_split
from sklearn.preprocessing import StandardScaler
import numpy as np
import matplotlib.pyplot as plt
```

```
def find_file(path):
    with open(path) as file:
        for line in file:
            yield line

def read_data(filename):
    X = []
    Y = []
    XX = []
    YY = []
    data = []
    for line in find_file(filename):
        content = line.strip().split()
        content = list(map(int, content))
        if content[-1] > 0:       #  >20
            data.append(content)

    size = len(data)
    rs = random.sample(range(0,size), int(size * 0.7))
    for item in rs:
        X.append(data[item][:-1])
        Y.append([data[item][-1]])

    for i in range(0, size):
        if i not in rs:
            XX.append(data[i][:-1])
            YY.append([data[i][-1]])
    return X, Y, XX, YY, rs, data

filename = 'hair.txt'
X_train, y_train, X_test, y_test, rs, data = read_data(filename)

X_scaler = StandardScaler()
y_scaler = StandardScaler()
X_train = X_scaler.fit_transform(X_train)
y_train = y_scaler.fit_transform(y_train)
X_test = X_scaler.transform(X_test)
y_test = y_scaler.transform(y_test)

regressor = SGDRegressor(loss='squared_loss')
scores = cross_val_score(regressor, X_train, y_train, cv=5)
print('Cross␣validation␣R␣squared␣value:', scores)
print('Cross␣verify␣the␣R␣squared␣mean:', np.mean(scores))
regressor.fit(X_train, y_train)
print('Test␣set␣R␣squared:', regressor.score(X_test, y_test))
print(regressor.coef_)
print(regressor.intercept_)

print(len(rs))
```

```
# predictions = regressor.predict(X_test)
# for i, prediction in enumerate(predictions):
# print('Predicted: %s, Target: %s' % (prediction, y_test[i]))
p_result = []
r_result = []
for i in range(364):
    if i not in rs:
        y_pre = regressor.coef_[0] * data[i][0] + regressor.coef_[1] * da
                + regressor.coef_[4] * data[i][4] + regressor.intercept_
        y_real = data[i][5]
        p_result.append(y_pre)
        r_result.append(y_real)
plt.plot(range(len(p_result)),p_result,color='red',label='predict')
plt.plot(range(len(r_result)),r_result,color='green',label='real')
plt.xlabel('The␣product_parent␣serial␣number')
plt.ylabel('Sales␣amount')
plt.title('Fitted␣value␣by␣our␣pattern')
plt.legend(loc=0)
plt.show()

# print(p_result)
# print(r_result)
```

Appendix C Third appendix

Here is part of python sentiment analysis programs we used in our model.

Input python programes:sentiment_process2.py

```
from textblob import TextBlob
import pandas as pd

def find_file(path):
    with open(path, 'r', encoding='UTF-8') as file:
        for line in file:
            yield line

def readFile(path):
    i = 0
    star = []
    help = []
    total = []
    for line in find_file(path):
        if i == 0:
            i=i+1
            continue
        content = line.strip().split('\t')
        if(len(content[0]) > 0 and len(content[1]) > 0 and len(content[2]
            if int(content[0]) == 1 :
                star.append(-5)
```

```
            elif int(content[0]) == 2 :
                star.append(-4)
            elif int(content[0]) == 3 :
                star.append(0)
            else:
                star.append(int(content[0]))
            help.append(int(content[1]))
            total.append(int(content[2]))
    return star, help, total

path = '486774008.txt'
star, help, total = readFile(path)
data = ""
test = pd.read_table(path)
clomn = test['review_headline']

for line in clomn:
    #line.replace(".",",")[-1]
    data = data + line.replace(".",",").replace("!","").replace("?","") +
blob = TextBlob(data)
sentences = blob.sentences
print(len(sentences))

print(len(star), len(help), len(total))
plority = []

for i in range(len(star)):
    plority.append(blob.sentences[i].polarity)
    print(star[i], help[i], total[i], blob.sentences[i].polarity)

print(len(plority))

p1 = 0.3 * sum(star) * 0.2

p2 = 0
for i in range(len(plority)):
    if plority[i] >= 0:
        p2 = p2 + plority[i] * (help[i] + 1) + (plority[i] - 1) * 0.5 * (
    else:
        p2 = p2 + plority[i] * (help[i] + 1) + (plority[i] + 1) * 0.5 * (

print(p1 +0.7 * p2)
```