Problem Chosen	2020 MCM/ICM Summary Sheet	Team Control Number
C		2014070

Gain a deep insight into online market

summary

Nowadays, shopping online gains its currency. An increasing number of consumers prefer to shopping in e-commerce website other than the hypostatic stores for the convenience. To make consumers aware of the quality of products, many websites credit the commodities using buyers' remarks. Therefore, to prompt the sales online, sellers have no alternative but to pay much attention to the reviews, also improving products according to those reviews.

In this paper, we find out relevant variables that may affect star ratings and give suggestions to Sunshine Company.

Initially, we build the framework called ICEF (identification, characterization, evaluation and forecast) for the rest of our work. To gain all measurable variables, we use **NLP** and other data processing methods to deconstruct the review into three variables: topic correlativity, emotional tendencies and length. After being applied **logical reasoning** and **Pearson correlation analysis**, variables that have strong relationships with stars and helpfulness votes come into light. Then, we combine **entropy method** and **decision tree model** to characterize the relationships within and between them. To give the Sunshine Company the most informative index to track, we pioneeringly construct 4 indexes utilizing different approaches. By comparing the strengths and weaknesses between the 4 indexes, we finally judge reputation function (i.e. index 4) has the most valuable information to the company. **The reputation function** integrates star-influential factors and review-influential factors, endowing them with different importance. We also formulate an **ARMA model** to analyze and predict the changing trend of reputation function. Moreover, to have a deep insight into the reviews, we apply **low-rating chain** and **correlation analysis** to reveal relationships within reviews.

At last, after analysis and prediction, we determine strategies that the Sunshine Company should apply to prompt sales if they want to place their products on sales on the internet. (1) Specific to the products, we have suggestions about the design. For example, to the pacifier, it should be not only cute and adorable, appealing to babies but also is convenient to wash. The pacifier is made up of good, natural rubber instead of plastic for the safety of babies. (2) The reputation of new products is higher than that of old, so the company should release new products continuously. (3) several products have excellent prospects should be noticed, such as samsung counter top microwave, panasonic hair dryer nano care…

We finally conduct sensitivity analysis, dissect advantages and disadvantages of our model and present a letter to the marketing director of Sunshine Company.

Keywords: online reviews, NLP, reputation function

Letter

To: market director of Sunshine Company

From: Team # 2014070

March **9, 2019**

Subject: online reviews' characterization, analysis and prediction

Honorable Mr./Mrs. in Sunshine Company,

Currently, online market is critically crucial to your company, so appealing to consumers online is the top priority on the agenda. So, our team analyzed the given data and conceived several models to characterize, evaluate and predict the reputation of one specific product. To better satisfy your requests, we perform sensitive analysis, exploring strengths and weaknesses of our models.

Results

We utilized entropy method and decision tree model to character the relationships within and between star ratings, helpfulness votes and reviews. The relationship between star ratings, emotional scores of reviews and population type is quite strong. From observing the dataset, we reveal several regular patterns about helpfulness votes. We pioneeringly determined the most valuable and informative index for your company to use. This reputation function takes multiple factors into consideration very seriously, which can help you to evaluate the product in the online market. In the future, we predict the reputation may increase with the time goes by.

Proposal

To improve the reputation of products and prompt the sales of Sunshine Company, we propose the following suggestions for each type of products.

Microwave:
- the speed should be fast enough
- material of microwave is stainless
- the lifespan of button is long, handling door is easy to open

Pacifier:
- cute and adorable, appealing to babies
- convenient to wash
- made up of good, natural rubber instead of plastics

Hair dryer:
- the speed of drying is fast
- long, retractable cords, easy to handle with
- making as less noise as possible

That’s our analysis, results and suggestions for your company.

Yours sincerely,

Contents

1 Introduction

With the prosperity of e-commerce, sellers put too much emphasis on online sales which have relationships with products' scores given by the website. By taking users' comments, star-ratings and helpfulness rating into account, the website grades the commodities using machine learning. Therefore, to boost sales, businesses are desired to know how these factors interact with each other and how they affect the product's reputation.

As network evaluation becomes under great concern, the same issue falls upon Sunshine Company that tries to introduce three new products (microwaves, hair dryers, pacifier) into the marketplace. Particularly, we are hired as consultants to address the requests:

- giving suggestions on their online sales strategy;
- identifying potentially important design features that would enhance product desirability.

In this work, we proceed in several phrases, including identification, characterization, evaluation and forecast. The following framework shows the logic of solving problems, presented in the following figure.

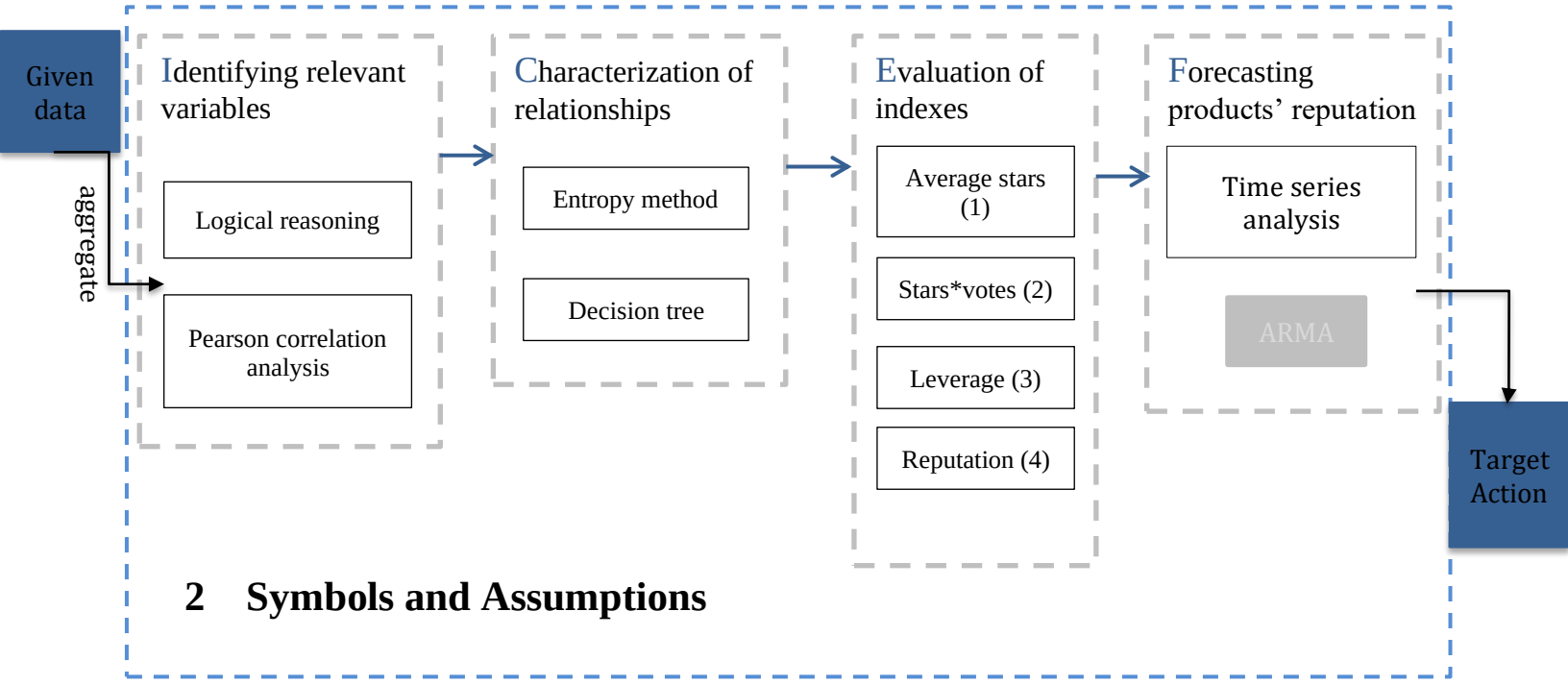
The framework can be explained by the next few steps:

Identification: We first use NLP to deconstruct reviews, splitting it to three variables (topic correlativity, emotional tendency, length). Then by applying logical reasoning and Pearson correlation analysis, we select relevant variables to the model.

Characterization: find out the relationship within and between star ratings, reviews and helpfulness votes with entropy method and decision tree.

Evaluation: We construct and compare four types of indexes, attempting to determine one measure that is the most informative.

Forecast: utilize time series analysis to reveal the changing trend of a product's reputation over time.



2 Symbols and Assumptions

2.1 Symbols

Symbol	Definition
SR	SR is the star ratings of the review
num	the total number of reviews for a product
HV	the number of helpful votes of a review
len	the length of reviews
pos	the positive tendency of reviews, $pos \in [0,1]$
neu	the neutral tendency of reviews, $neu \in [0,1]$
neg	the negative tendency of reviews, $neg \in [0,1]$

2.2 General Assumptions

Assumption1: *Assume that all the data is true.*

Despite the incompleteness of the dataset and some errors in statistics, we make this assumption to guarantee one valid solution.

Assumption2: *"Online water army" don't exist to affect the credibility of all information*

Rules of e-commerce websites are easy to be seriously damaged by "water army", and their profusion of fake information.

3 Analysis of data

First, we use Excel to open the .tsv file by the coding of UTF-8. Then, after discovering that some data which is not associated with the data set by searching product title, we delete the irrelevance data and get 3 files: pure_microwave.xlsx, pure_hair_dryer.xlsx, pure _pacifier.xlsx.

3.1 Quality the reviews

The reviews contain copious information, but it's unrealistic to read and analyze them one by one due to that not all the information is valuable. To obtain usable information, we adopt NLP (natural language processing) to analyze these reviews.

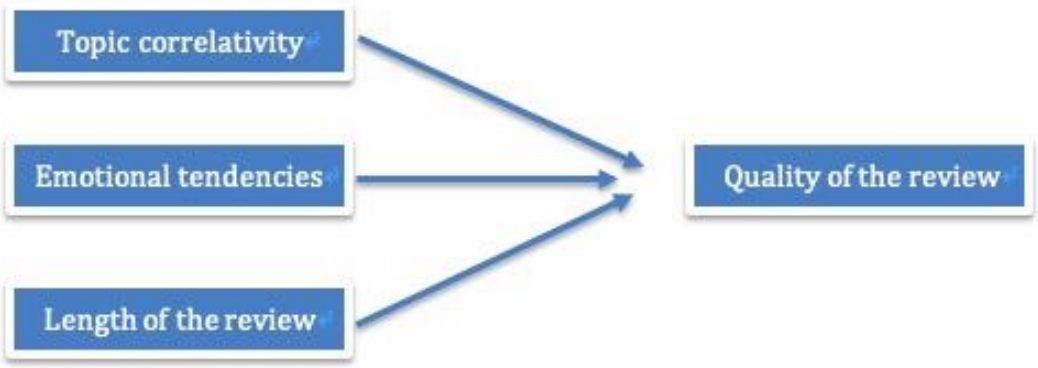


figure 3.1.1 Quality of the review

Rating the quality of reviews from three aspects: topic correlativity, emotional tendencies and length. Previous studies show that researchers find the most important features of online reviews about products are topic correlativity, emotional tendencies and length of the review.

3.2 Emotional tendencies

A knotty problem is to extract users' attitudes towards products by dissecting the bodies of reviews. Besides that, one also need to quantify emotional tendencies which means attaching emotional score to every review.

What we choose right here is natural language processing (NLP) to rate the emotional score of every comment. The core ideas of this method are to cut each word in the text, to qualify it, and then compare it with the word bank, deriving emotional tendencies.

Then we get emotional tendencies, and they can be divided into three classes: positive, neutral and negative.

$$pos + neu + neg = 1 \qquad (1)$$

$$pos \in [0,1] \;\; neu \in [0,1] \;\; neg \in [0,1] \qquad (2)$$

Pos means positive emotion tendency, neu means neutral emotion tendency, and neg means negative emotion tendency.

To a large extent, the emotional accuracy of the analysis depends on the accuracy and richness of the corpus. Thus, we use Corpus of Contemporary American English (COCA) and NLTK as word banks and process in Anaconda.

After getting the result, we notice that it is not very precise. For example, a review (review ID: RO9KNA385PZML in hair dryer's data set) writes: 'No complaints.' However, the negative point of it is 1, which is completely oppose the real situation. Thus, we adjust data modestly based on the following rules:

(1) If possessing high star rating, pos should gain more credits than neg; If the review has low star rating, pos should be smaller than its neg;

(2) Generally, the value of $(pos - neg)$ of reviews and the star ratings of the reviews have the same order.

(3) Preserve the integrity of data

The details of this adjustment's process can be seen in 'Change.gms' in Appendix B.

Before we adjust the result, we select out some reviews that should not be changed. For example, a review (review ID: R3DIF90R3C5WHO in hair dryer's data set) that has 4-stars rating writes: 'Too noisy.' and get a negative point of 0.63. We think this kind of reviews should be preserved.

3.3 Topic correlativity

Only when the review is related to our topic, this review would be useful and helpful. To measure the topic correlativity of a review, we use word2vec method.

First of all, we manually select the frequently used and theme related words from all reviews of a product(whose detail is in 'find_the_words.m' in Appendix B), and we store these theme related words in a vector.

Then, for each specific review, we extract the real words, counting the occurrence frequency of these real words in the theme related word vector.

We calculate the weighted average of these occurrence frequencies to get the topic correlativity score.

3.4 length of the review

From the previous researches, we found that reviews' length is an essential indicator of review. The longer a review, the more detailed it is, and more helpful to others, so we must pay attention to the importance of comments' length.

Obviously, the reviews' length is relatively easy to obtain. We only need to count the number of words in the review.

3.5 relationship between specific descriptors and rating levels

Under the context of different star ratings, we look for more frequent real words. Then, we actually find some specific quality descriptors strongly associated with rating levels.

The strongly descriptors for each product are listed below.

Hair dryer	quiet (high rating)	strong (high rating)	heavy (low rating)
microwave	fast (high rating)	stainless (high rating)	broken (low rating)
pacifier	adorable (high rating)	washable (high rating)	plastic (low rating)

table 3.5.1 strongly descriptors for each product

These feedbacks can help designers to understand consumers' thoughts and wishes, then improving qualities of products.

4 Part 1: Relationship Characterization

From the dataset, we find some relationships between the variables through some Quantitative or qualitative data mining methods.

1. Types of reviewers and emotional tendency have a strong influence on star rating.

2. Helpful votes' numbers of a review are impacted by the review's topic correlativity and date.

We then will illustrate these two relationships.

4.1 Symbols and Assumptions in Section 4

Symbol	Definition
N_t	the number of samples in decision leaf t
$E(t)$	the information entropy of leaf t
T_{leaf}	the number of leaf node in the tree

4.2 Relationship One

As we have mentioned, types of reviewers and emotional tendency have a strong influence on star rating.

4.2.1 Preliminary screening and logical analysis

At first, after observation of data, we select effective factors among others: qualities of reviews, population type.

The reasons why getting rid of other variables are that marketplace is uniform and almost every id (including customer id, review id) is different. That is to say, taking them out of consideration has no effect on our analysis of data.

Besides, we find out correlations between other aspects such as topic correlativity, length, helpfulness votes and topic correlativity with star-rating are very weak. (table 4.2.1.1)

The correlation coefficient between star-rating and others				
types of products	**helpfulness vote**	**total vote**	**length of review**	**theme of review**
hair dryer	**-0.0734**	**-0.1206**	**-0.0918**	**-0.0452**
microwave	**0.0113**	**0.0006**	**-0.1714**	**-0.1111**
pacifier	**-0.045**	**-0.0595**	**-0.1169**	**0.0047**

table 4.2.1.1 The correlation coefficient between star-rating and others

In our intuitive logical analysis, the types of reviewer and the emotional tendency of the text will affect the star rating.

4.2.2 decision tree model

We will use decision trees to find relationships between star rating and review type, emotional tendency. Decision tree is a theory of machine learning, which is mainly used in classification and data mining.

First, we divide star rating [1,2,3,4,5] into five classes as *labels*, we want to explore what kind of feature that each *label* (i.e. star rating) corresponds to.

Lables(star rating)	1	2	3	4	5

table 4.2.2.1

According to types of reviewer, we classify reviewers into three categories:

(1) vine;

(2) non-vine but verified purchaser;

(3) non-verified.

As the stem mentions, reviewers in amazon are different that can be attributed to whether they purchase products in amazon or not, whether they buy discounted merchandises or not. we assume these differences may also influence their star-ratings.

Based on division of reviewer type, we use $feature1$ [0,1,2] to symbolize corresponding type.

$$feature1 = \begin{cases} 0 & vine \\ 1 & purchaser \\ 2 & nonpurchaser \end{cases}$$

Next, because review emotional tendency reflects the reviewer's feeling, this feeling will influence star rating.

Here, we define $feature2$ as the feeling score.

$$feature2 = feeling\ score = pos - neg$$

where pos is positive tendency, neg is negative tendency, and when feeling score is high, it indicates the review's feeling is more positive.

After proper and precise data preprocessing and defining labels and features, we apply decision tree to explore the relationship between star ratings, feeling scores of reviews and population type.

Here we introduce information entropy, which reflects degree of disorder.

$$E(t) = \sum_{i \in t} -p_i log p_i$$

Where t is a leaf, i is the element of t, $E(t)$ is entropy of leaf t, pi is possibility of i.

In the decision tree, one uses information entropy in search of optimal dividing, aiming at minimizing the entropy values. There are many types of samplings in the training data set. By judgement of information entropy, equipment can divide the data set by layers, separating out each type of sample.

The evaluation function of decision tree is

$$C(T) = \sum_{t \in leaf} N_t E(t)$$

In this equation, $C(T)$ is the evaluation function of tree, and smaller $C(T)$ is, the better the tree is. Nt is the sample number in leaf t. $E(t)$ is the entropy of leaf t.

Meanwhile, one need to avoid the occurrence of overfitting, that is a hypothesis about the training data to obtain better fitting than the other hypothesis but can't fit other data outside the training data as well. Therefore, tree pruning is must for better results. Stopping the construction of the tree by early pruning, presetting the evaluation to be minimized is a must.

Thus, we improve the evaluation function as

$$C_\alpha(T) = C(T) + \alpha |T_{leaf}|$$

Where $C\alpha(T)$ is the modified evaluation of decision tree, $|T_{leaf}|$ is the total number of leaf nodes in this tress, and α is coefficient.

At last, after several rounds of operation, the algorithm stops after an accurate and uniform partition of the data set is formed (that is, the star rating of the data can be predicted more accurately by using the tree). At this time, the prediction accuracy of the model is over 70%. We generate some visualized graphs of trees to make them intuitional. (figure 4.2.2.1)

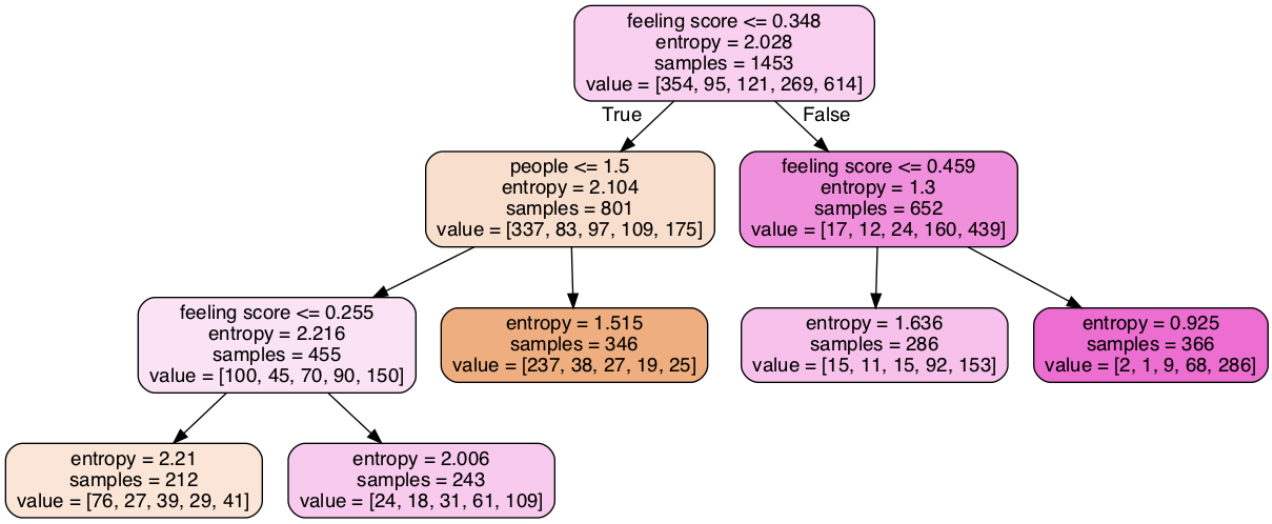


figure 4.2.2.1

The results show that at first, people are divided into three categories (vine, no vine with verified purchase and no vine without verified purchase). Then, under each category, the algorithm illustrates the relationship between star-ratings and emotional tendencies.

1.Those people classified as no vine without verified purchase give low ratings

Explanation: products in amazon have quality assurance that guarantee customers buy certified mechanizes in amazon. Thus, customers are very likely to purchase goods of low quality and then assume the products are very bad.

2.There is positive correlation between emotional score and ratings

Explanation: After having a deep insight of the bodies of reviews, one can find out that people tend to write a lot of praise, thinking the product is successful. While, when they consider products as awful, they just truthfully describe reasons why they think so. In this case, their reviews are more neutral, less negative.

4.2 helpfulness ratings

Judging from common sense, we build the relationship between helpfulness ratings and the quality of reviews. As mentioned before, the quality of reviews determined by three aspects: topic correlativity, emotional tendency and length. After counting the Pearson correlation coefficient between these values, however, one can find that the relationships between those three aspects and helpfulness ratings are not linear. (table4.2.1)

Pearson correlation coefficient			
types of commodities	length	topic correlativity	Emotional tendencies
microwave	0.370	0.310	-0.0198
Hair dryer	0.282	0.192	-0.0680
pacifier	0.220	0.146	-0.750

table 4.2.1 Pearson correlation coefficient

What's more, after checking the data carefully, one realizes that finding an explicit relationship between them seems impossible, and machine learning can't give us a satisfied result, as well.

Reasons why finding out the quantified relationships between them is unreasonable are just as followings:

- Amazon always keeps the high-votes reviews on the top, meaning that the customers will see those reviews first. Since the high-votes reviews are high-quality, the possibility of customers' appreciating them is high, which make their votes become more and some good reviews go unrewarded.
- The numbers of different products' buyers are not the same, meaning that helpfulness votes are correlated with products of different brands. Also, it means that, to hot-sale products, excellent reviews are more likely to be overlooked and the helpfulness votes of high-vote reviews may be overestimated.

Based on reason 1, the relationship between helpfulness ratings and the quality of reviews cannot be linear relationship. Based on reason 2, talking about the helpfulness ratings without a certain product is meaningless. Nevertheless, if focusing on a certain product, one can conclude that:

- ❒ the earlier the comment, the more likely to receive more helpfulness votes. (see figure 4.2.1)
- ❒ the longer the comment is written, the more relevant it is to the topic, and the more probable it is to be highly praised. (judging from the following figures)
- ❒ If there is a high praise response within a relatively short period of time from a person's comments, unless the person's response quality is very high, it is possible that the number of likes will be reduced. (see figure 4.2.2)

The following figures are two kind of microwaves, showing the relationship among helpfulness vote, the quality of the review and the date, where the white circles represent the helpfulness vote, the blue bar represents the quality of the review and the x axis represents the date

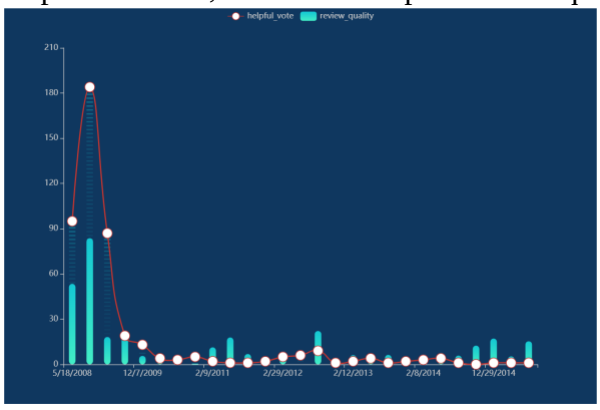


figure 4.2.1

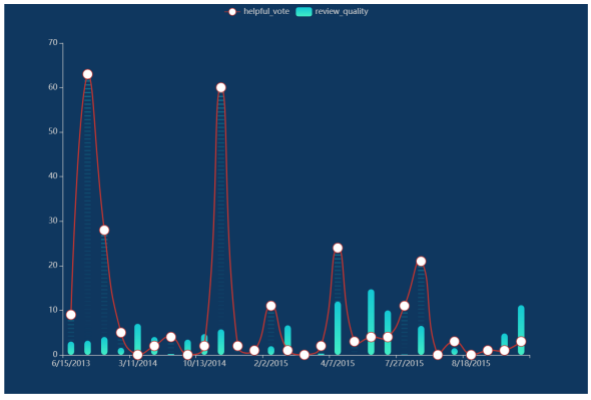


figure 4.2.2

5 Part 2: Analysis, description and forecasting

In this section, we solve the problem a and problem c first, since the relationship between them is really strong. Then, we will use the reputation function and time series to forecast products'

reputation in the future, which addresses problem b. We define low-rating chain to describe the data set, using correlation analysis to solve the final problem.

5.1 Symbols in Section 5

Symbol	Definition
Pop	the popularity of a specific product,
n	the value of time interval
$distance$	the length of time interval
ESR	the effect of star ratings
EHV	the effect of helpful votes
EP	the effect of this people
EL	the effect of the review's length
Re	Reputation of a product
enu	the enthusiastic score of a review
dsa	the disappointed score of a review

5.2 Identify data measures

The sunshine company needs to track the most informative ratings and reviews, so what we are expected to find out the valuable index which embodies ratings and reviews. We put forward four indexes to evaluate products.

We think the best index should include the following properties:

1. This index needs to be quantifiable.
2. The best index should be informative and contain as much information as possible.

5.2.1 Average – score index1

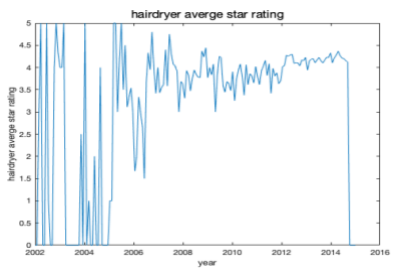
The simplest way to evaluate good or bad is the average number. For example, when a teacher wants to assess the learning level of a class, the first thing is to see the average score of this class.

In this idea, we obtain $score\ index1$.

$$score\ index1 = \frac{\sum_1^{num} SR}{num}$$

In this equation, *num* means the total number of reviews, SR is the star ratings of the review, and $score\ index1$ is the average.

This measure fits into the logic and is easy to understand. By average star ratings, sellers can have a general idea about whether the majority of buyers appreciate the products or not. The company utilizes a concise but objective index to know about their consumers. Following graphs show the results of this measure.



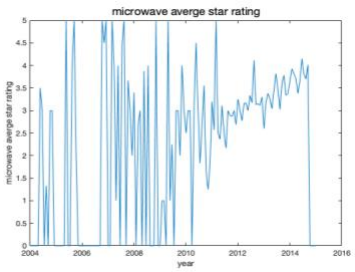


Figure5.2.1.1 microwave average Figure 5.2.1.2 pacifier average Figure5.2.1.3 hair dryer average

From these graphs, one can learn that in earlier years, there are large fluctuations in the average star rating no matter the product type is. Between 2011 to 2014, the average star rating becomes stable. During the earlier periods, the number of buyers purchasing the product isn't large, so the index is sensitive to all star ratings. If there is one 5 star or 1 star, it can influence the result dramatically. With the increase of the number, average star rating gradually becomes insensitive to every rating. However, this approach has its own limitations.

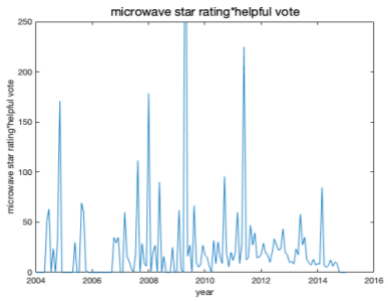
5.2.2 Star times vote - score index2

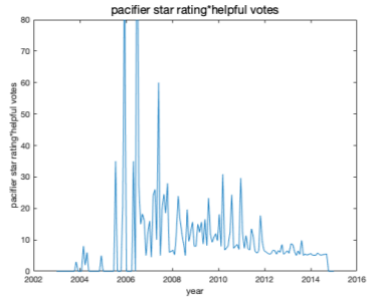
The $score\ index1$ has many drawbacks including that it can't reflect the importance of every comment. By considering a phenomenon of malicious swiping, it is unreasonable to think that average star rating is the most informative index. Therefore, we come up to $score\ index2$, which adds the HV (helpful votes) into our model.

$$score\ index2\ = \frac{\sum_1^{num}(SR \times HV)}{num}$$

In this equation, SR is the star ratings of the review, HV is the number of helpful votes, and helpful vote*s* resemble a weight which measures the importance of star rating. Therefore, we can identify which star rating is important. For example, if there are two comments A and B, when A's star, B's star, A' helpful votes, B's helpful votes are respectively 5,4,10,20, the $score\ index2 = (5 \times 10 + 4 \times 20) \div 2 = 65$.

Now, different results are indicated by the graphs.





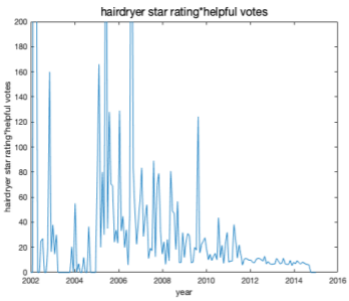


Figure5.2.1.4 microwave average Figure 5.2.1.5 pacifier average Figure5.2.1.6 hair dryer average

Judging from the graphs, there is fluctuations in almost every period of time especially in 2006-2010. During 2012-2014, the curve becomes smooth and the index gradually falls down to zero. From this aspect, sellers can gain less information by using these graphs which show no visible trend, so the index can't guide their production.

5.2.3 Leverage - score index3

There are limitations of the first two method, so making efforts to improve the index is must to us. By browsing the amazon website, we find something about Amazon's scoring algorithm.

Figure5.2.1.7 Amazon's scoring method

We infer from this photo, for a specific product, the scoring algorithm may be relevant to star rating, helpful votes, the reviews' date, the length of reviews, the total num of reviews.

The length of a review will influence the importance of this review. We find the longer the review is, the more influential it is. In order to measure the effect of length, we take the form of logarithm.

$$EL = log_5(len)$$

El is the effect of the review's length, and len is the length of a review.

For example, when review1's length is 125 and review2's length is 5, from this equation, the effect of review1 is 3 times as review2. After many tries, we find base as 5 is suitable in the logarithm.

For a specific product, the total number of reviews can measure this popularity. If there are a lot of reviews for a product, it shows many people are aware of this product and this product is popular. To measure the effect of review number, we come up

$$Pop = \sqrt{num}$$

Pop means the popularity of a specific product, and *num* is the review number.

The review date also has a significant influence on the scoring algorithm. The older the review is, the less likely it is to be seen by other consumers, which leads this review to be less important. We describe the time distance in this way

$$distance = (n+1)^2$$

$distance$ means time interval, symbolizing the importance of a review. The importance is inversely proportional to distance. For instance, this year is 2020. The review's distance in 2020 is 1. If a review is in 2018, its *distance* is 9, and its influence will become 1/9 times as the same review in 2020.

For different reviewers, their review's influence is different. We classify reviewers into three categories (1) vine; (2) non-vine but verified purchaser; (3) non-verified.

$$EP = \begin{cases} 2 & vine \\ 1 & purchaser \\ 0.5 & nonpurchaser \end{cases}$$

EP means the effect power of this type of people. Vine is invited by Amazon, so they have the biggest impact. Purchasers in Amazon have more influence than non-purchasers.

The impact of helpful votes is direct and credential. Base on this thought, we get

$$EHV = HV$$

EHV symbolize the effect of helpful votes, HV is helpful votes of a review.

As for star rating, we find out it has a bimodal distribution. 1star rating and 5star rating are the majority. We think 1-2 star rating is negative rating, so it also has negative effect to the scoring. In the same way, 4-5 star rating is positive, and has positive effect to the scoring. Thus, we get

$$ESR = SR - 3$$

SR is the star ratings of the review. ESR means the effect of star rating. More specifically, the effect of 1star rating is -2, the effect of 5star rating is 2. We describe it in this form

SR	1	2	3	4	5
effect	-2	-1	0	1	2

Considering all the variables and effect, we put forward an important and informative $score\ index3$.

$$score\ index3 = (\sum_{i=1}^{num} \frac{ESR_i \times EHV_I \times EL_i \times EP_i}{distance_i}) \times \frac{1}{num} \times Pop$$

where i is the i^{th} review for a specific product, and *num* is the total review number of the product. For the i^{th} review, ESR_i , EHV_i, EL_i ,EP_i are the effect of its star rating, effect of its helpful vote, effect of its length and effect of people. *Pop* means this product's popularity, because $Pop = \sqrt{num}$

The score index3 can also be expressed in

$$score\ index3 = (\sum_{i=1}^{num} \frac{ESR_i \times EHV_i \times EL_i \times EP_i}{distance_i}) \times \frac{1}{\sqrt{num}}$$

This model physical meaning is that ESR(effect of star rating) will have a positive or negative effect on the score. In addition, EHV(effect of helpful vote), EL(effect of review length) and EP(effect of people) resemble a leverage and amplify the ESR. And *distance* has a shrinking effect on the ESR.

The $score\ index3$ is similar to a leverage, and we also call it "leverage index".

$Score\ index3$ has contained a lot of information, having a powerful explanation to illustrate whether a product is successful and potential or not.

5.1.4 Reputation - score index4

Now we need to determine combination of text-based measures and rating-based measures that best indicate a potentially successful or failing product, we pioneeringly introduce $score\ index4$ as this best combination.

As we know the positive mood review will increase the score, and the negative mood review will decrease the score, and we define feeling influence as following:

$$EF = pos - neg$$

Where EF means the effect of feeling, pos and neg symbolize Emotional tendencies and vary from 0 to 1. For example, when $pos = 1$ and $neg = 0$ in a review, $EF = 1 - 0 = 1$. When $pos = 0.3$ and $neg = 0.7$ in another review, this time, EF is $0.3 - 0.7 = -0.4$ and has a negative effect

With a combination of EF, we can modify $score\ index3$ and get $score\ index4$,

$$score\ index4 = (\sum_{i=1}^{num} \frac{(w_1 \times ESR_i + w_2 \times EF_i) \times EHV_i \times EL_i \times EP_i}{distance_i}) \times \frac{1}{\sqrt{num}}$$

Here, num is the total review number and is the same as the num in $score\ index3$. W_1 and W_2 are the weights between ESR (effect of star rating) and EF (the effect of feeling).

This model physical meaning is that the combination of ESR and EF will have a positive or negative effect on the score. In addition, EHV, EL and EP resemble a leverage and amplify the effect combination of ESR and EF. And distance has a shrinking effect on the combination.

According the rule that the best index should be quantifiable and most informative, $score\ index4$ well meet these requirements and indicate a potentially successful or failing product. We will use $score\ index4$ as a measure of reputation to forecast the product development.

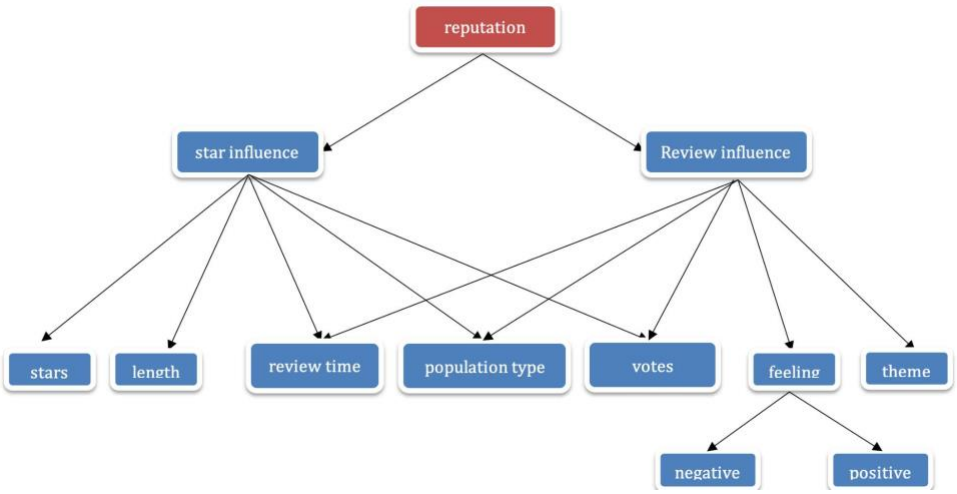


Figure5.2.1.8 The relationship between the factor

5.2 Time series

We use *score index4* as reputation function to describe a product's development. We find that reputation has a strong relationship with past reputation, which is consistent with the characteristics of time series. In this part we will use ARMA model to forecast how a product's reputation is increasing or decreasing in the online marketplace.

5.2.1 Overall observation

Before we analyze each specific product, we decide to observe hair dryer, microwave, pacifier class overall reputation situation, which will absolutely help us understand the market tendency.

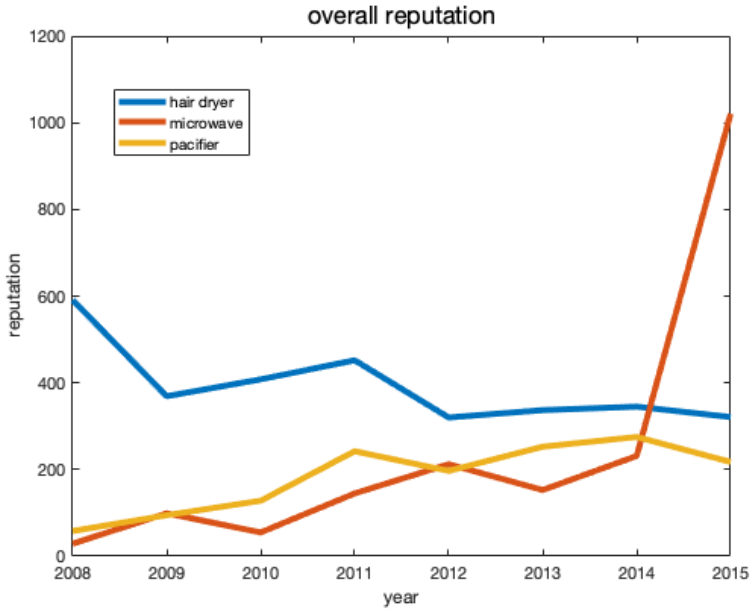


Figure 5.2.1.1 overall reputation

From the figure, we can find that the hair dryer overall reputation has slightly decreased, pacifier has increased slowly, and the microwave reputation is soaring.

5.2.2 ARMA model forecast

For specific product, according to time correlation, we decide to use ARMA model to forecast its reputation tendency.

$$Re_t = \beta_0 + \sum_{i=1}^{p} \beta_i Re_{t-i} + \epsilon_t + \sum_{i=1}^{q} \alpha_i \epsilon_{t-i}$$

Where Re_i is reputation in i year, ϵ_i is the error term in i year, β and α are Corresponding coefficient.

Take overall reputation for example, we determine the p and q by ACF and PCAF. The following figure respond hair dryer, pacifier, and microwave respectively.

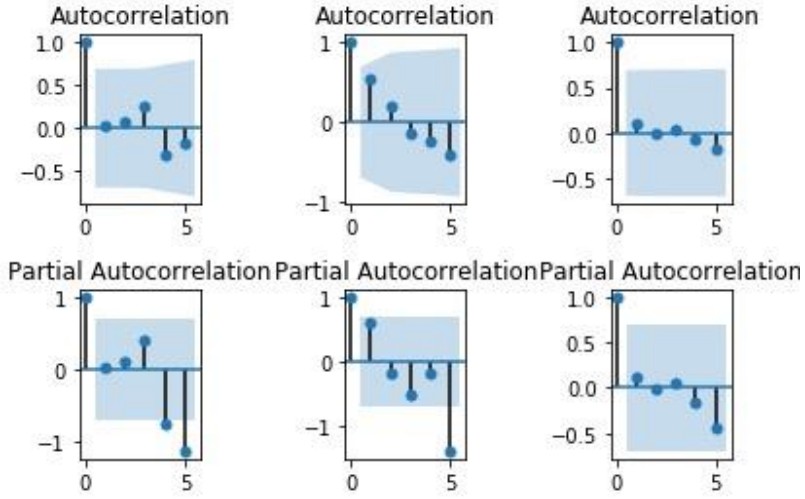


figure 5.2.2.1

After analyzing many specific products, we find that when lag q=1and p=1, it may have the best prediction effect and accuracy.

5.2.3 Forecast result and sensitivity analysis

We use the reputation from 2009 to 2014 to model the reputation tendency and utilize the reputation in 2015 to examine our model's accuracy. For most product, the results and accuracy are satisfactory.

However, for minority of all product, the result is not so good. We find the main reason for this problem is that this product is new product and issues in 2014 or 2015. There are not enough reputation data for ARMA to fit, and AMRA is sensitive to reputation data sample numbers, so the fitting result is not adapting. Therefore, we have to evaluate these new products separately.

After our work, we obtain a conclusion that new product's reputation generally is better than the old product, because the new product contains more advantages.

In the following form, we list some potential and successful products'parents.

Hair dryer	127431946	646149518	391944105
Microwave	692404913	464779766	423421857
Pacifier	905342430	812583172	450475749

table 5.2.3.1 successful products'_parents.

5.4 Question d: Using low-rating chain to analysis

After giving a definition of low-rating chain, we are sure that some people indeed prefer to write a good-sentiment review after seeing a series of low star ratings.

Before defining low-rating chain, some steps used to process the data is necessary. First, separate the data set by different products, since one will not write a good-sentiment review for

'product A' after seeing a series of low star ratings of 'product B'. Second, sort the review by date, making it easier to use the order of chart to instead of the order of date.

5.4.1 The definition of low-rating chain

A series of data which is consistent and has the same order of date in a chart is called low-rating chain, if

(1) It contains one high-rating data (4 stars or 5stars) at most, and the average rating of it is lower than 2;

(2) The length of it is not lower than 3, unless there is no data before it;

(3) If there is data before or after it, the data should be a high-rating review, which may have a rating of 5 stars or a rating of 4 stars.

(4) It can only contain one another low-rating chain at most. If there is no data after it, it can not contain a high-rating review.

The followings are some examples.

order	rating
A1	1
A2	1
A3	1
A4	1
A5	1
A6	5
A7	1
A8	5

chart A

order	rating
B1	1
B2	1
B3	5
B4	1
B5	1
B6	5
B7	1
B8	1

chart B

order	rating
C1	1
C2	1
C3	1
C4	5
C5	1
C6	1
C7	1
C8	5

chart C

order	rating
D1	5
D2	5
D3	1
D4	1
D5	5
D6	1
D7	1
D8	1

chart D

There are two low-rating chains in chart A, A1-A5 and A1-A7. There are two low-rating chains in chart B, B1-B2, and B1-B5. There are also two low-rating chains in chart C, C1-C3, and C5-C7. C1-C7 is not a low-rating chain, since it contains two low-rating chains and based on the fourth rule. In cart D there is only one low-rating chain, D6-D8, and D3-D8 is not a low-rating chain because D5 is high-rating.

From the definition of the low-rating chain, one can find that there are basically two kinds of low-rating chains. The first is that there is a high rating data after the chain, which means a customer may give a good-sentiment review after seeing so many low star ratings. The reason why we can not confirm this person indeed gives a good-sentiment review is that we find some 4-stars review which shows the dissatisfactory and disappointment. The second is that there is not a high rating data after the chain, meaning that no one wanted to give a good-sentiment review after this series of low star ratings.

Next, we will use low-rating chain to analysis microwaves' data set.

5.4.2 Low-rating chains in microwaves' data set

With the definition of low-rating chains, one can find that there are 17 products in microwave's data set that contains 43 low-rating chains. We read the reviews after the low-rating chains carefully (if there is a review after it). For example, a customer wrote a review (review ID: R2GMLIROXE3WZU in microwaves' data set) with title "GE Profile --4 stars for now looks only!", and give 4-stars rating. We don't think this kind of review represents a good sentiment.

Another problem is that how can we determine that the reason of customer wrote the review is that seeing a series of low star ratings, since the low-rating chain may show that the relationship between giving good-sentiment review and a series of low star ratings is strong, but it can not demonstrate that the customer had been influenced by the low star ratings. In order to solve this problem, we check the review again and find out 5 reviews (review ID: R3MHYSFID-E9OBA, R245MU1ZCJQ77G, R17H2H5K17JF45, R3AZNMFSZ6IC7N, R3U37SQ3U1QGOI) that **directly** writing that they had seen the low star ratings and all of them are after low-rating chains. Thus, the assumption that a good-sentiment review after a low-rating chain indicates that the customer had been influenced by the low star ratings is reasonable.

Finally, we find there are 36 good-sentiment reviews after the low-rating chains, which is 83.7% of the total. Hence, we come to the conclusion that some people indeed prefer to write a good-sentiment review after seeing a series of low star ratings.

5.5 Descriptors of text-based reviews associated with rating levels

To describe whether a text is 'enthusiastic' or not need sentiment analysis and emotional detection, which will be difficult if the emotion is complex. In order to describe the emotion 'enthusiastic', we associate it with pos, neu, and neg, which have been calculate above.

First, we assume that if a review is described as enthusiastic, it will be positive in some degree. Furthermore, due to the strong passion of the reviewer, the neu of an enthusiastic review should be lower. Hence, we use $pos - 0.5 \times neu - neg$ as the score of enu which can describe whether a review is enthusiastic. Then, we calculate the correlation coefficient between enu and SR and got the following result:

product	correlation coefficient
microwave	0.903
pacifier	0.368
hair dryer	0.423

table 5.5.1 correlation coefficient between enu and SR

From the correlation coefficient we can learn that high rating and enu is indeed positively correlated, but the linearity between them is not very strong.

Similarly, we can define the score of dsa as $neg - 0.5 \times neu - pos$ and the correlation coefficient just become negative but the absolute value of it isn't change.

Finally, we come to a conclusion that enthusiastic or disappointed review may have a strong relationship with rating level, but this kind of relationship should not be linear.

6 Strength and Weakness

6.6.1 Strength

1. We identify the relationship, measures, and parameters by a combination of quantitative and qualitative patterns;
2. We construct a best score index as the standard of measurement. This index well meets the informative and effective requirement;
3. Our ARMA model is consist with reputation data development tendency and volatility.

6.6.2 Weakness

1. The emotional analysis of review can't be able to as accurate as the top company which specialize in NLP;
2. ARMA is hard to predict some new product, due to inadequate review. That is to say, we have to separate some new product to make additional analysis;
3. The method used in part 2 problem e is not very precise.

7 Conclusion

Ratings and reviews given by consumers are credential to the company who sells products online. In this report, we characterized the relationship within and between star ratings, helpfulness votes and reviews, using entropy method and decision tree model. After comparing with other indexes, we determine that the reputation index (i.e. index 4) is the most informative to the users, also forecasting the changing trend of the reputation. Besides that, we define low-rating chain which indicates that a series of low-star rating increases the probabilities of writing good-sentiment reviews.

Reference

[1] Forman, Chris, Anindya Ghose, and Batia Wiesenfeld. “Examining the Relationship Between Reviews and Sales: The Role of Reviewer Identity Disclosure in Electronic Markets.” *SSRN Electronic Journal*, 2007. https://doi.org/10.2139/ssrn.1026893.

[2] Ghose, A., and P. G. Ipeirotis. “Estimating the Helpfulness and Economic Impact of Product Reviews: Mining Text and Reviewer Characteristics.” *IEEE Transactions on Knowledge and Data Engineering*23, no. 10 (2011): 1498–1512. https://doi.org/10.1109/tkde.2010.188.

[3] Sompras, Gamgarn, and Pattarachai Lalitrojwong. “Extracting Product Features and Opinions from Product Reviews Using Dependency Analysis.” *2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery*, 2010. https://doi.org/10.1109/fskd.2010.5569865.

Appendix A

1 A part of nlp result:

microwave:

product_parent	star_rating	neg	neu	pos	theme	length
109226352	1	0.274834	0.663166	0.062	0.08356	43
109226352	1	0.344222	0.603778	0.051	3.69154	232
109226352	1	0.293	0.707	0	0	7
109226352	1	0.370063	0.629937	0	0	22
109226352	2	0.38044	0.51156	0.108	1.494513	69
109226352	2	0.389592	0.555408	0.056	5.978052	174
109226352	2	0.399561	0.530439	0.07	0.458485	37
109226352	2	0.41117	0.52983	0.06	2.293164	44
109226352	3	0.066	0.81	0.124	2.25619	220
109226352	3	0	0.528	0.472	0.182654	11
109226352	4	0.235	0.376856	0.389144	0	11
109226352	4	0.079	0.484436	0.436564	3.871979	90
109226352	4	0.035	0.4727	0.4923	1.957435	45
109226352	4	0.024	0.529424	0.445576	1.142515	99
109226352	4	0.015	0.574412	0.410588	1.927855	78
109226352	4	0	0.558112	0.441888	3.342503	137
109226352	5	0.118	0.495788	0.386212	0.24625	82
109226352	5	0.113	0.48501	0.40199	3.6028	137
109226352	5	0.085	0.47233	0.44267	8.62839	122
109226352	5	0.083	0.332216	0.584784	1.84355	16

hair dryer:

product_parent	star_rating	neg	neu	pos	theme	length
694290590	1	0.372	0.628	0	0	7
694290590	1	0.372	0.5338	0.0942	0	15
694290590	1	0.567936	0.432064	0	0.43092	29
694290590	1	0.711748	0.288252	0	0	8
694290590	2	0.47754	0.370252	0.152208	0.117783	13
694290590	2	0.489224	0.364212	0.145564	0	20
694290590	2	0.396	0.48018	0.12382	2.253263	45
694290590	2	0.437676	0.443336	0.118988	0.253141	41
694290590	3	0.031	0.844	0.125	4.64983	74
694290590	3	0	0.78	0.22	0.77717	40
694290590	3	0	0.865	0.135	0.2599	21

694290590	3	0.069	0.909	0.022	18.4168	135
694290590	4	0.065032	0.621499	0.313469	10.46312	183
694290590	4	0.064293	0.54686	0.389847	0.710587	77
694290590	4	0.061337	0.560162	0.378501	2.146295	20
694290590	4	0.060598	0.424186	0.516216	0.106968	20
694290590	5	0.035	0.389	0.576	0.78528	23
694290590	5	0.0335	0.4175	0.548	0.99415	44
694290590	5	0.0325	0.331	0.6365	2.30413	39
694290590	5	0.0325	0.2805	0.687	0.2114	25

pacifier:

product_parent	star_rating	neg	neu	pos	theme	length
293975317	1	0.257	0.743	0	1.3135	16
293975317	1	0.319412	0.680588	0	0.13684	27
293975317	1	0.257	0.743	0	0.06118	29
293975317	1	0.318669	0.681331	0	1.60102	29
293975317	2	0.785342	0.487482	0.127982	0.434736	22
293975317	2	0.594145	0.552192	0.042421	0.780838	36
293975317	3	0.113	0.887	0	0.786242	28
293975317	3	0	0.918	0.082	1.945826	44
293975317	3	0	0.802	0.198	0.73479	21
293975317	4	0.0435	0.379	0.5775	0.7897	49
293975317	4	0.037	0.397	0.566	1.04436	42
293975317	4	0.017	0.452	0.532	0.48683	64
293975317	4	0.016	0.4085	0.5765	1.637122	80
293975317	4	0.012	0.42	0.568	0.744088	134
293975317	4	0	0.367	0.633	0.907519	33
293975317	4	0	0.3675	0.6325	2.083315	58
293975317	5	0.020672	0.476544	0.501784	2.98067	119
293975317	5	0.01632	0.320416	0.663264	1.01993	38
293975317	5	0	0.184416	0.815584	0.73479	6
293975317	5	0	0.403104	0.596896	1.08824	13

2 A part of reputation evaluate result

microwave:

	215953885	242727854	295520151	305608994	309267414
2004	0	0	0	0.201763244	0
2005	0	0	0	314.4783556	0
2006	0	0	0	78.62939683	0
2007	0	0	0	50.03519302	0
2008	0	-301.0497452	0	47.47901572	0

2009	0	-75.34907753	395.9578126	20.27087566	0
2010	0	-41.90138768	118.4172175	20.03690098	0
2011	0	-15.82639017	75.66198645	30.93832988	0
2012	79.88167006	-16.5095364	35.07041622	27.62776846	0
2013	9.864668528	-18.27590567	36.52304561	24.96638902	0
2014	-0.322835329	-14.6205372	-16.78747014	22.99893192	9.7762663
2015	3.189379328	-8.415137097	54.53602523	9.853851837	7.66751824
2016	24.18	-12.16	27.49	15.56	4.98
2017	18.95	-13.5	36.07	18.66	4.05

hair dryer:

	108191918	109106777	121009604	122140779	127343313
2002	0	0	0	0	0
2003	0	0	0	0	0
2004	0	0	0	0	0
2005	0	-120.9686035	0	0	0
2006	0	-25.7044707	0	0	0
2007	299.7246993	-12.3065359	0	0	0
2008	74.93117484	-7.056351034	0	86.01059285	0
2009	33.30274437	-7.699838419	0	43.04773455	0
2010	49.40381886	-1.490882036	0	22.82321258	70.9593586
2011	26.94157686	79.63211918	0	17.71247091	65.2386326
2012	34.512128	28.54645749	14.75548343	13.83336743	179.878104
2013	84.7941164	27.78742246	19.25496958	10.29270541	59.1349075
2014	42.69498185	30.11914232	14.17675473	12.85852607	206.450155
2015	27.54098919	26.09067117	10.18139851	9.557786856	150.240334
2016	45.89	39.72	11.54	12.42	127.86
2017	43.77	36.88	11.59	12.83	143.02

pacifier:

	28435092	33280098	44176469	51308962	51313971
2003	0	0	0	0	-5.06245801
2004	0	0	0	0	-1.03586735
2005	0	0	0	0	16.5316461
2006	0	0	0	0	39.187257
2007	0	0	0	0	14.6604476
2008	0	0	0	0	8.46247979
2009	0	0	0	0	3.89147386
2010	0	0	0	2.580384292	2.35348259

2011	0	0	0	0.645096073	1.59425413
2012	0	0	13.25795187	6.855261779	1.15559406
2013	7.670833245	3.243203715	10.03405189	4.240213491	0.87741288
2014	13.75993076	6.900720582	7.60028108	1.442254871	0.68938523
2015	16.47370097	7.831120597	8.743061331	2.27348914	0.55615841
2016	14.77	6.98	6.94	3.14	0.69
2017	13.38	6.3	6.53	3.02	0.79

Appendix B

feelinganalyze.py

```
import csv
import nltk
from nltk.sentiment.vader import SentimentIntensityAnalyzer

alldata=[]

def nltkSentiment(view):
    sid = SentimentIntensityAnalyzer()
    #view_sen=[]
    for sen in view:
        #print(sen)
        feeling=[]
        senti = sid.polarity_scores(sen)
        for k in senti:
            # print('{0}:{1},'.format(k, senti[k]), end='\n')
            number1=senti[k]
            meizu.append(number1)
        alldata.append(feeling)

txt1=[]

with open('paciffeel.csv','r',encoding='UTF-8') as f:
    reader = csv.reader(f)
    for line in reader:
        try:
            c = line[0].strip()
            txt1.append(c)
        except:
            b='error'
            txt1.append(c)



nltkSentiment(txt1)
```

```
file=open('mood analyze.csv','a',newline='')
content=csv.writer(file,dialect='excel')
for i in alldata:
   content.writerow(i)
file.close()
```

Change.gms

```
option optcr=0
option optca=0
option lp=cplex;
option threads=8;
option ITERLIM=10000000;
option ResLiM=200000000;
sets
n  sentiment value /n1*n3/
p  product        /p1*p1594/
s  star           /s1*s5/
;

parameters
va(p,n)
ss1
ss2
ss3
ss4
ss5
;

ss1=398;
ss2=510;
ss3=643;
ss4=935;
ss5=1595;

parameter va(p,n)   the value of sentiment
$call GDXXRW microwave_change.xlsx trace=3 par = va rng=sheet1!a1 rdim=1 cdim=1
$GDXIN  microwave_change.GDX
$LOAD va
$GDXIN
;

variables
a(s)
b(s)
obj
```

```
;
positive variables

ae(s)
be(s)
;

equations
eq1(p)
eq2(p)
eq3(p)
eq4(p)
eq5(p)
eq6(p)
eq7(p)
eq8(p)
eq9(p)
eq10(p)

eq11(p)
eq12(p)
eq13(p)
eq14(p)
eq15(p)
eq16(p)
eq17(p)
eq18(p)
eq19(p)
eq20(p)

eq21(p)
eq22(p)
eq23(p)
eq24(p)

eqq1
eqq2

eeq1(s)
eeq2(s)


eeq3
;

eq1(p)$(ord(p)<ss1)..      va(p,'n1')+a('s1')*va(p,'n2')=l=1;
```

```
eq2(p)$(ord(p)<ss1)..      va(p,'n1')+a('s1')*va(p,'n2')=g=0;
eq3(p)$(ord(p)<ss2 and ord(p)>ss1-1)..      va(p,'n1')+a('s2')*va(p,'n2')=l=1;
eq4(p)$(ord(p)<ss2 and ord(p)>ss1-1)..      va(p,'n1')+a('s2')*va(p,'n2')=g=0;
eq5(p)$(ord(p)<ss3 and ord(p)>ss2-1)..      va(p,'n1')+a('s3')*va(p,'n2')=l=1;
eq6(p)$(ord(p)<ss3 and ord(p)>ss2-1)..      va(p,'n1')+a('s3')*va(p,'n2')=g=0;
eq7(p)$(ord(p)<ss4 and ord(p)>ss3-1)..      va(p,'n1')+a('s4')*va(p,'n2')=l=1;
eq8(p)$(ord(p)<ss4 and ord(p)>ss3-1)..      va(p,'n1')+a('s4')*va(p,'n2')=g=0;
eq9(p)$(ord(p)<ss5 and ord(p)>ss4-1)..      va(p,'n1')+a('s5')*va(p,'n2')=l=1;
eq10(p)$(ord(p)<ss5 and ord(p)>ss4-1)..      va(p,'n1')+a('s5')*va(p,'n2')=g=0;

eq11(p)$(ord(p)<ss1)..      va(p,'n3')+b('s1')*va(p,'n2')=l=1;
eq12(p)$(ord(p)<ss1)..      va(p,'n3')+b('s1')*va(p,'n2')=g=0;
eq13(p)$(ord(p)<ss2 and ord(p)>ss1-1)..      va(p,'n3')+b('s2')*va(p,'n2')=l=1;
eq14(p)$(ord(p)<ss2 and ord(p)>ss1-1)..      va(p,'n3')+b('s2')*va(p,'n2')=g=0;
eq15(p)$(ord(p)<ss3 and ord(p)>ss2-1)..      va(p,'n3')+b('s3')*va(p,'n2')=l=1;
eq16(p)$(ord(p)<ss3 and ord(p)>ss2-1)..      va(p,'n3')+b('s3')*va(p,'n2')=g=0;
eq17(p)$(ord(p)<ss4 and ord(p)>ss3-1)..      va(p,'n3')+b('s4')*va(p,'n2')=l=1;
eq18(p)$(ord(p)<ss4 and ord(p)>ss3-1)..      va(p,'n3')+b('s4')*va(p,'n2')=g=0;
eq19(p)$(ord(p)<ss5 and ord(p)>ss4-1)..      va(p,'n3')+b('s5')*va(p,'n2')=l=1;
eq20(p)$(ord(p)<ss5 and ord(p)>ss4-1)..      va(p,'n3')+b('s5')*va(p,'n2')=g=0;

eq21(p)$(ord(p)<ss1)..      va(p,'n1')+a('s1')*va(p,'n2')=g=va(p,'n3')+b('s1')*va(p,'n2');
eq22(p)$(ord(p)<ss2 and ord(p)>ss1-1)..
va(p,'n1')+a('s2')*va(p,'n2')=g=va(p,'n3')+b('s2')*va(p,'n2');
eq23(p)$(ord(p)<ss4 and ord(p)>ss3-1)..
va(p,'n1')+a('s4')*va(p,'n2')=l=va(p,'n3')+b('s4')*va(p,'n2');
eq24(p)$(ord(p)<ss5 and ord(p)>ss4-1)..
va(p,'n1')+a('s5')*va(p,'n2')=l=va(p,'n3')+b('s5')*va(p,'n2');

eqq1..  sum(p$(ord(p)<ss1), va(p,'n3')+b('s1')*va(p,'n2')-va(p,'n1')-a('s1')*va(p,'n2'))/(ss1-
1)=l=sum(p$(ord(p)<ss2 and ord(p)>ss1-1), va(p,'n3')+b('s2')*va(p,'n2')-va(p,'n1')-
a('s1')*va(p,'n2'))/(ss2-ss1);
eqq2..  sum(p$(ord(p)<ss4 and ord(p)>ss3-1), va(p,'n3')+b('s4')*va(p,'n2')-va(p,'n1')-
a('s4')*va(p,'n2'))/(ss4-ss3)=l=sum(p$(ord(p)<ss5 and ord(p)>ss4-1),
va(p,'n3')+b('s4')*va(p,'n2')-va(p,'n1')-a('s4')*va(p,'n2'))/(ss5-ss4);

eeq1(s)..      ae(s)=g=a(s)-b(s);
eeq2(s)..      be(s)=g=b(s)-a(s);


eeq3..  obj=e=sum(s,ae(s)+be(s));

model change /all/;
solve  change using lp min obj;
display obj.l, a.l, b.l, va;
```

find_the_words.m

```
% start to find words in a text
ch = fileread('test.txt');
ch = lower(ch);
str = replace(ch,{'.', '?', '!', ',', ';', ':', '<br />'},' ');
str = split(str);
tbl = tabulate(str);
tbls = sortrows(tbl, 2, 'descend');
```

draw.html

```
<!DOCTYPE html>
<html>
<head>
   <meta charset="utf-8">
   <title>ECharts</title>
</head>
<body>
   <script type="text/javascript" src="echarts.js"></script>
        <script type="text/javascript" src="data.js"></script>
   <div id="main" style="width: 900px;height:600px;"></div>
   <script type="text/javascript">
      var myChart = echarts.init(document.getElementById('main'));
                //lg over-the-range microwave oven with 300 cfm venting system
                //profile 2.2 cu. ft. countertop microwave w/child lockout and extra large



                // option
                option = {
                        backgroundColor: '#0f375f',
                        tooltip: {
                                trigger: 'axis',
                                axisPointer: {
                                        type: 'shadow'
                                }
                        },
                        legend: {
                                data: ['helpful_vote', 'review_quality'],
                                textStyle: {
                                        color: '#ccc'
                                }
                        },
                        xAxis: {
                                data: category,
                                axisLine: {
                                        lineStyle: {
```

```
                        color: '#ccc'
                    }
                }
            },
            yAxis: {
                splitLine: {show: false},
                axisLine: {
                    lineStyle: {
                        color: '#ccc'
                    }
                }
            },
            series: [{
                name: 'helpful_vote',
                type: 'line',
                smooth: true,
                showAllSymbol: true,
                symbol: 'emptyCircle',
                symbolSize: 15,
                data: lineData
            }, {
                name: 'review_quality',
                type: 'bar',
                barWidth: 10,
                itemStyle: {
                    barBorderRadius: 5,
                    color: new echarts.graphic.LinearGradient(
                        0, 0, 0, 1,
                        [
                            {offset: 0, color: '#14c8d4'},
                            {offset: 1, color: '#43eec6'}
                        ]
                    )
                },
                data: barData
            }, {
                name: 'line',
                type: 'bar',
                barGap: '-100%',
                barWidth: 10,
                itemStyle: {
                    color: new echarts.graphic.LinearGradient(
                        0, 0, 0, 1,
                        [
                            {offset: 0, color: 'rgba(20,200,212,0.5)'},
                            {offset: 0.2, color: 'rgba(20,200,212,0.2)'},
```

```
                                                {offset: 1, color: 'rgba(20,200,212,0)'}
                                            ]
                                        )
                                },
                                z: -12,
                                data: lineData
                        }, {
                                name: 'dotted',
                                type: 'pictorialBar',
                                symbol: 'rect',
                                itemStyle: {
                                        color: '#0f375f'
                                },
                                symbolRepeat: true,
                                symbolSize: [12, 4],
                                symbolMargin: 1,
                                z: -10,
                                data: lineData
                        }]
                };

                myChart.setOption(option);
    </script>
</body>
</html>
```

reputation.m

```
clc
clear
%prepare to caculate the reputation
%the difficulty is to filled the default value
all_product = xlsread('hair_dryer_reputation_processing.xlsx', 1, 'A2:K11113');
product_kind = xlsread('hair_dryer_reputation_processing.xlsx', 1, 'L2:L127');
num = size(product_kind, 1);

small_year = 2002;
large_year = 2015;
yenum = large_year - small_year + 1;

product_reputation = zeros(num*(large_year - small_year + 1), 4);
for pi = 1:num
        position = find(all_product(:, 1) == product_kind(pi,1));

        time = all_product(position(1,1):position(size(position,1),1), 5);

        star = all_product(position(1,1):position(size(position,1),1), 2) - 3;
```

```
vote = all_product(position(1,1):position(size(position,1),1), 3:4);
votein = 2*vote(:, 1) - vote(:, 2) + 1;

vine = all_product(position(1,1):position(size(position,1),1), 7);
vp = all_product(position(1,1):position(size(position,1),1), 8);

review_len = all_product(position(1,1):position(size(position,1),1), 6);
re_influence=log(review_len + 1)/log(5);

feeling = all_product(position(1,1):position(size(position,1),1), 10)...
        - all_product(position(1,1):position(size(position,1),1), 9);

theme = all_product(position(1,1):position(size(position,1),1), 11);

year = small_year;
k = 1;
flag = 0;
for i = 1:yenum
    product_reputation((pi-1)*yenum + i, 1) = product_kind(pi,1);
    if(time(k,1) ~= year && flag == 0)
        product_reputation((pi-1)*yenum + i, 2) = year;
        product_reputation((pi-1)*yenum + i, 3) = 0;
        if(k == 1)
            product_reputation((pi-1)*yenum + i, 4) = 0;
        else
            for u = 1:i
                product_reputation((pi-1)*yenum + i, 4) =
product_reputation((pi-1)*yenum + i, 4)...
                +product_reputation((pi-1)*yenum + u, 3)/(i-u+1)/(i-u+1);
            end
        end
        year = year + 1;
    elseif(time(k, 1) == year && flag == 0)
        pos = find(time == year);
        if((pos(size(pos, 1), 1) + 1) <= size(time, 1))
            k = pos(size(pos, 1), 1) + 1;
        elseif(time(k, 1) < large_year)
            flag = 1;
        end
        product_reputation((pi-1)*yenum + i, 2) = year;
        for u = 1:size(pos, 1)
            product_reputation((pi-1)*yenum + i, 3) = product_reputation((pi-
1)*yenum + i, 3)...
```

```
                                                + (0.7*star(pos(u, 1),1)*votein(pos(u,
1),1)*re_influence(pos(u, 1),1) + 0.3*feeling(pos(u, 1),1)*theme(pos(u, 1),1)*votein(pos(u,
1),1))...
                                                *(vine(pos(u, 1),1)+vp(pos(u, 1),1));
                        end

                        product_reputation((pi-1)*yenum + i, 3) = product_reputation((pi-
1)*yenum + i, 3)/sqrt(size(pos, 1));

                        for v = 1:i
                                product_reputation((pi-1)*yenum + i, 4) = product_reputation((pi-
1)*yenum + i, 4)...
                                +product_reputation((pi-1)*yenum + v, 3)/(i-v+1)/(i-v+1);
                        end
                        year = year + 1;
                        continue;
                end
                if(flag == 1)
                        product_reputation((pi-1)*yenum + i, 2) = year;
                        product_reputation((pi-1)*yenum + i, 3) = 0;
                        for u = 1:i
                                product_reputation((pi-1)*yenum + i, 4) = product_reputation((pi-
1)*yenum + i, 4)...
                                +product_reputation((pi-1)*yenum + u, 3)/(i-u+1)/(i-u+1);
                        end
                        year = year + 1;
                end
        end
end
```

forecast.py

```
import csv
import statsmodels
from statsmodels.graphics.tsaplots import *
import matplotlib.pyplot as plt
from statsmodels.tsa import arima_model

alldata=[]
with open('hair dryer.csv','r',encoding='UTF-8-sig') as f:
   reader = csv.reader(f)
   for line in reader:
      shuju=[]
      for i in line:
         data.append(i)
      model1=arima_model.ARIMA(data,order=(1,0,1)).fit()
      forecast=model1.forecast(2)[0]
```

```
        forecast1=[round(x,2) for x in jieguo]
        alldata.append(forecast1)

file=open('hair dryer forecast.csv','a',newline='')
content=csv.writer(file,dialect='excel')
for i in alldata:
    content.writerow(i)
file.close()
```