| Problem Chosen | 2021 MCM/ICM Summary Sheet | Team Control Number |
| --- | --- | --- |
| C | | 2104744 |

# Crack the Market: Analysis of Amazon Review and Rating


**Summary**

In today's world, e-shopping is becoming more and more prevalent among the masses. Due to the missing physical experience, the reputation and quality of products in online shopping malls are more determined by user ratings and reviews. Therefore, merchants can collect this information to develop strategies for their sales.Based on reviews and ratings, we propose the Iteration-based Commodity Reputation Redistribution Model First, we mathematically modeled review confidence and overall product evaluation as targets. We split review confidence into reviewer confidence and review content confidence.

For reviewer confidence, we consider the Pearson coefficient between customer ratings and product ratings as the primary factor of customer confidence after considering the effect of instability of a few data ratings and treat it accordingly. For the confidence of the review content, we use the comprehensive characteristics of the reviews as variables, determine the weights using the FAHP evaluation method, and then determine the credibility of the review content by the TOPSIS method. After adding the review timeliness, we enhance the robustness through an iterative mechanism of reputation reassignment. Finally, we obtain the final composite score of the product, which is used to help Sunshine Company in market tracking.

For problem c, we choose a time-series approach to make predictions on the composite scores. The final results show that the market for all three commodities will continue to rise in the future.

For questions d and e, we used the correlation coefficient to represent the association between the two variables. With the final results obtained, there is a relationship between the review star rating and the review text content.

Last but not the least, we have built robust, interpretable and transferable models to assist Sunshine in market analysis and to give predictions for the future, which are essential for online market strategy development.


**Keywords**: Reputation Redistribution; correlation coefficient; review confidence

# Contents

# 1 Introduction

## 1.1 Background

Digital networks for product information have redefined traditional word-of-mouth social networks by allowing consumers to easily share their opinions and experiences with other members of large-scale online communities. Many online retailers, such as Amazon.com, are augmenting their product markets by building online communities to provide product reviews to other consumers[2]. Such information sharing not only has the potential to reduce the uncertainty consumers face regarding the quality of a product or a seller but also helps market seller make their strategies as a reference.

## 1.2 Problem Summary

Sunshine Company plans to introduce and sell three new products in the online marketplace: a microwave oven, a baby pacifier, and a hairdryer. We will use previous ratings and review data from other competitors to build our model to help Sunshine Company analyze market formats and develop sales strategies. The detailed work is listed below:

- Analyze the dataset of three products by statistical means to establish different measures that can reflect the reputation of the product.
- Based on the above measures, we will establish a time-based mathematical pattern to evaluate the comprehensive reputation of the product considering all aspects.
- Based on the established composite scoring pattern, Build models that can predict the future of products.
- Calculate the correlation coefficient between a specific rating and some specific review type to determine if there is some relationship between them.
- Calculates the correlation coefficient between a particular descriptor and a rating level to determine whether a relationship exists between them.
- Write a letter to the Marketing Director of Sunshine Company to summarize our teams analysis and results, and give appropriate advice on product sales strategy.

## 1.3 Our Model

The building process of our model is shown in the figure below

First, we clean the raw data set and use word search to remove reviews that do not belong to the target product. We defined several measures that can reflect product reputation, analyzed the relationship between average rating, number of reviews, word frequency, and product type, time, rating stars, and visualized them employing line graphs and word cloud graphs.

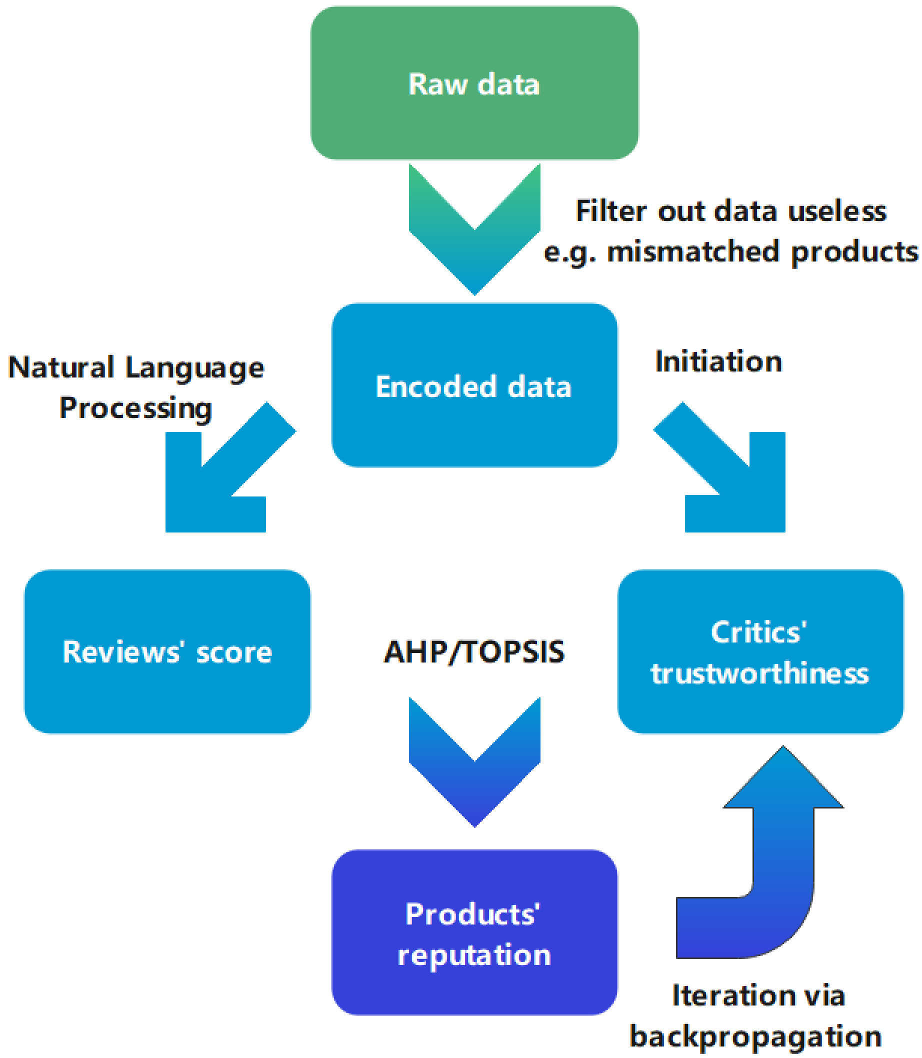


Figure 1: Flow Chart

Based on a binary network of users and products, we mathematically model the confidence of reviews and the overall product rating as objectives. We divide the review confidence into the reviewer's confidence and the confidence of review contents.

For the former, we use the Pearson coefficients between customer ratings and product ratings as the main factors of customer credibility after considering the effects of the instability of a few data ratings and processing them accordingly.

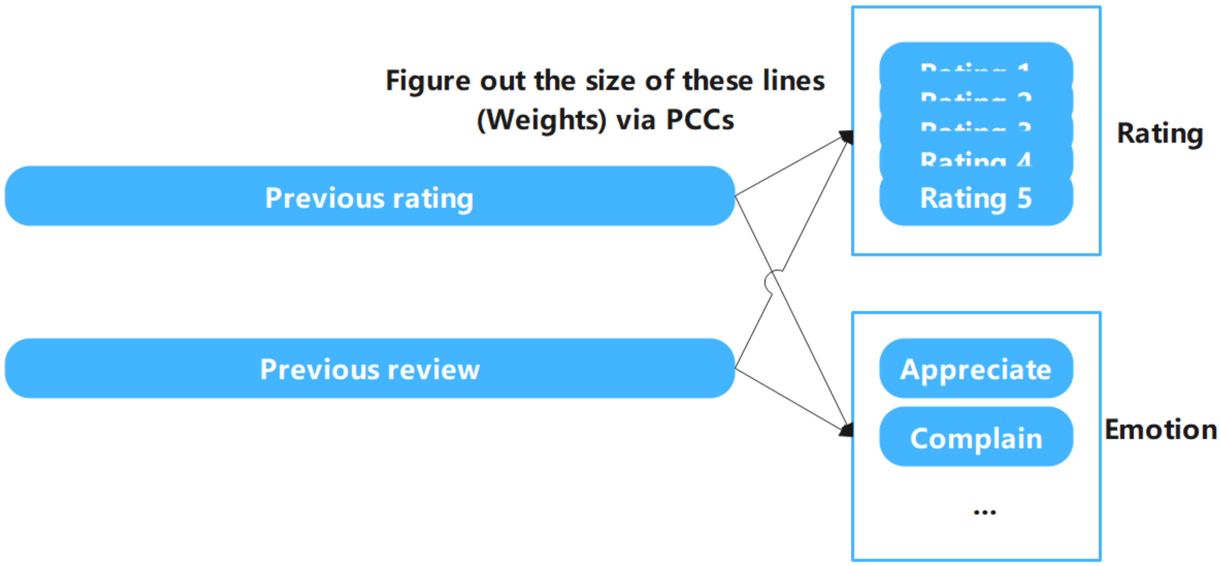


Figure 2: Combine features

For the latter, we use the external and content features of reviews as variables, determine the weights using the FAHP evaluation method, and then determine the confidence level of review content through the TOPSIS method.

After integrating the review timeliness, we enhanced the robustness through an iterative mechanism of reputation reassignment. Finally, we obtain the final composite rating of the product.

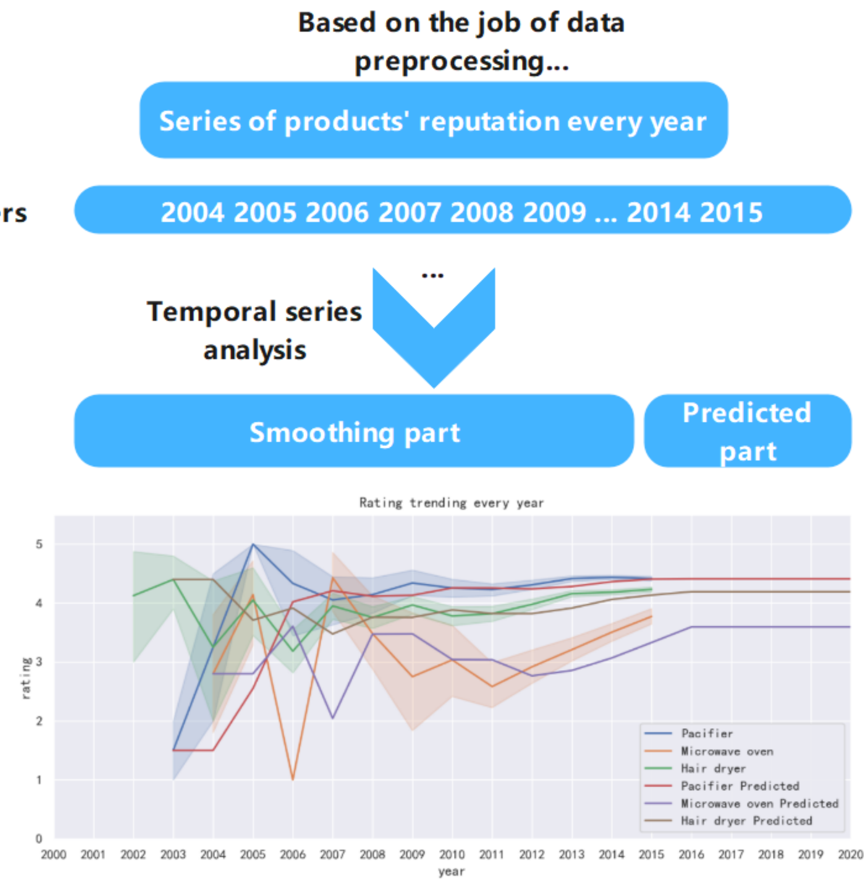


Figure 3: Predict future score

After obtaining the distribution of product reputation scores on the time axis, we will use a time-series approach to fit the reputation curve and make predictions about future reputation scores.

For (d) and (e) of Problem 2, we will calculate Pearson's correlation coefficients between a specific star rating, specific quality descriptors, and a review sentiment intensity, rating level, to determine whether there is some relationship between them.

Finally, we will write a letter to the Marketing Director of Sunshine Company to summarize our teams analysis and results, and give appropriate advice on product sales strategy.

# 2 Data Process and Insight

## 2.1 Data Cleaning

In the process of viewing the data, we found that the data set was not pure. In addition to our target products, it also contains other kinds of products that belong to the same large category. Therefore, we improve the purity of the data by retrieving whether the feature $product\_title$ contains pre-defined keywords to eliminate the reviews that do not belong to the target product.The table below shows the comparison of the number of comments before and after the cleaning.

| review number | hair-dryer | microwave | pacifier |
|---|---|---|---|
| before | 11471 | 1616 | 18938 |
| after | 11112 | 1604 | 11052 |

## 2.2 Data Mining

Initially, we analyzed the variation of the average rating over time.

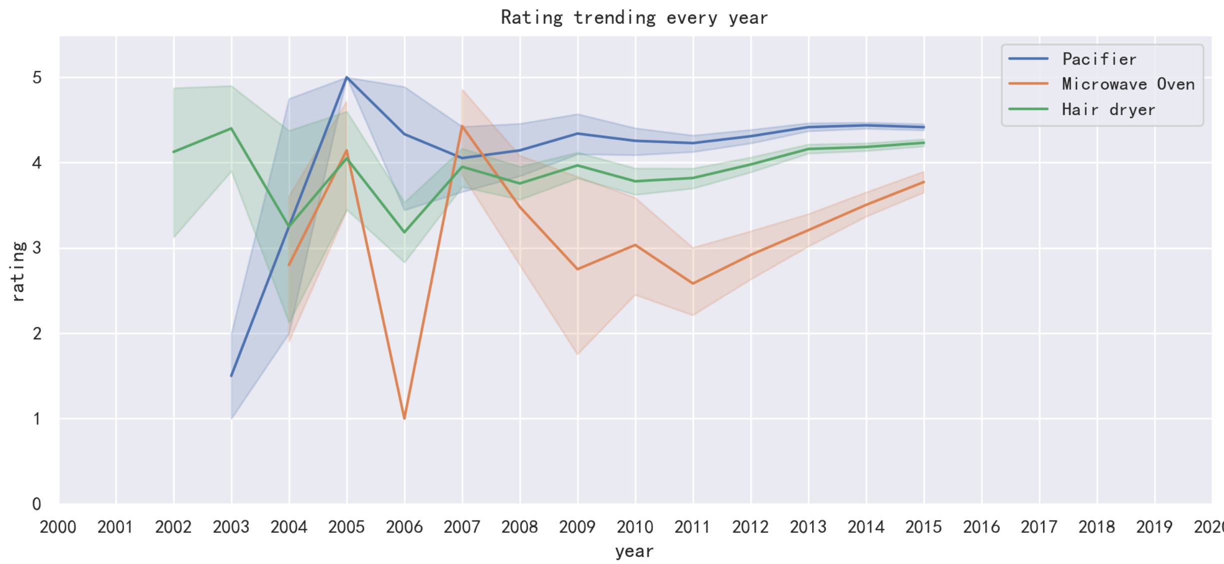


Figure 4: Average rating per year

From Figure 4, we can see that the average ratings of the three products fluctuate a lot in the beginning, have an increasing trend in general, and tend to converge to the same rating in the end.

Finally, we mined the reviews' contents, counted the high-frequency words in the three datasets, and created the corresponding word cloud maps.


Figure 5: Pacifier word map

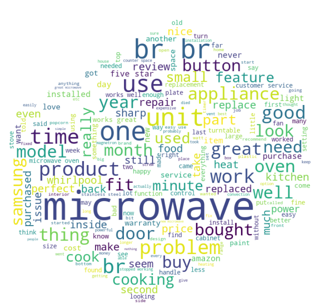


Figure 6: Microwave word map


Figure 7: Hair-dryer word map

In addition, we also analyzed the product categories in the three products, and the distribution of product categories is shown in the figure below.

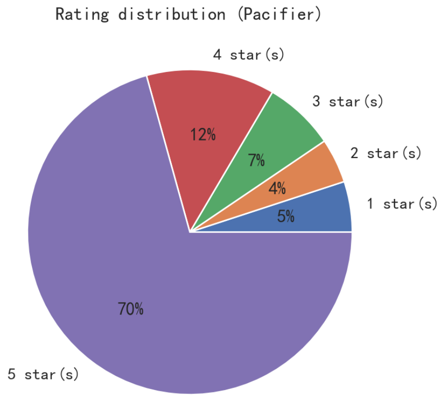


Figure 8: Pacifier star distribution

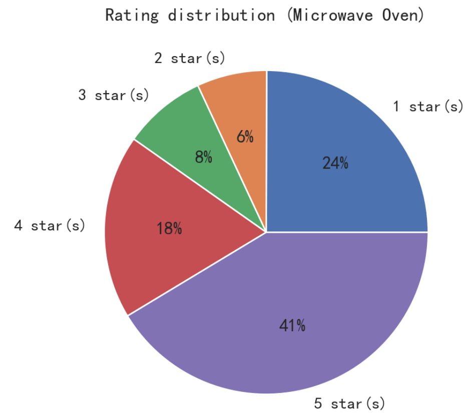


Figure 9: Microwave star distribution

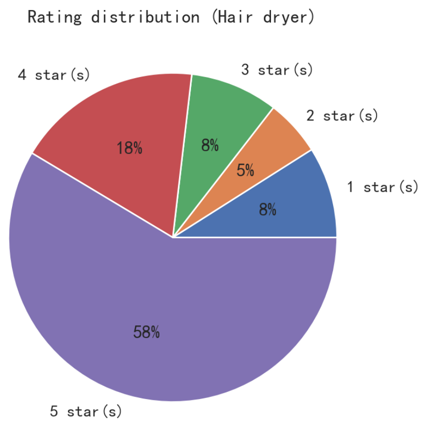


Figure 10: Hair-dryer star distribution

From the above star rating distribution chart can be seen, pacifier and hair dryer five-star rating accounted for the majority, while the microwave oven five-star and one-star rating accounted for the majority. So when your team is ready to enter the microwave oven market, be aware of the polarized word of Mouth.

# 3 Assumptions and Notations

## 3.1 Assumptions

To simplify our modeling, we make the following assumptions:

- There is a positive correlation between sales and the number of reviews.
- The content of the comments is true and unbiased

- The provided dataset contains all comments without extensive missing
- Individual differences between customers are ignored, such as economic, aesthetic and other factors

## 3.2 Notations

Table 1: Notation Table

| Symbol | Description |
|---|---|
| $H_1$ | the ratio of helpness and total votes |
| $H_2$ | the degree of emotion expression |
| $H_3$ | attribute word ratio |
| $H_4$ | proportion of adjectives |
| $H_5$ | the depth of review |
| $H_6$ | whether is vine |
| $H_7$ | whether is veried customer |
| $O$ | the set of all categories of goods |
| $N$ | the number of set O |
| $W$ | the confidence of review. |
| $r$ | the given rating of review |
| $C$ | the set of all customers |
| $w$ | the confidence of reviewer |
| $\delta_i$ | the confidence of the review content |
| $t$ | the timeliness of review |
| $\theta$ | a tunable parameter, which have an influence on reputation redistribution |
| $TR$ | the Pearson correlation coefficience |

# 4 Iteration-based Commodity Reputation Redistribution Model

Our comprehensive product evaluation model is built based on review confidence. Therefore, we will define two important metrics, which are product composite rating$Q$ and review confidence$W_i$.

## 4.1 the Defination of Commodity Reputation

The most straightforward method to rank objects is to consider their average ratings (we refer it as the mean method). However,such methods are very sensitive to the noisy information andmanipulation. In these rating systems, some users may give unreasonable ratings because they are not serious about the ratingor simply not familiar with the related field.Thus we refer to the method come up with the past article[4], finally define the quality and reputaton of good $Q$ as:

$$Q = \frac{\sum\limits_{j \in O} Q_j}{N}$$

$$Q_j = max\left\{W_i\right\}\frac{\sum r_i * W_i}{\sum W_i}(i \in C)$$

where:

- $O$ denotes the set of all categories of goods.
- $N = |O|$,denotes the number of set O.
- $W_i$ denotes the confidence of review $i$.
- $r_i$ denotes the given rating of review $i$.
- $C$ denotes the set of all customers.

Since different reviewers and contents would have an impact on the trustworthiness of the reviews themselves, we introduce the concept of review confidence. Compared to a simple rating average that defines the quality of an object, we assign a certain weight to each review in the calculation of the rating, which greatly increases the accuracy of the quality rating[1].

Also if an object is rated by one or two users, though the ratings are high, it is too arbitrary to claim this object has high quality[4].Also, in our hypothesis, we believe that there is a positive correlation between sales and the number of reviews.Therefore, we added penalty factors to the original weighted average equation. In turn, the ratings of products with few reviews are reduced.

## 4.2 the Defination of Customer Reputation

There is no doubt that the confidence of a comment should be composed of the confidence of the commenterthe confidence of the comment content and the timeliness[2]. Therefore, we define the comment confidence as:

$$W_i = w_i * \delta_i * t_i(i \in C_\alpha)$$

where:

- $w_i$ denotes the confidence of reviewer.
- $\delta_i$ denotes the confidence of the review content.
- $t_i$ denotes the timeliness of review.
- $C_\alpha$ denotes the set of good $\alpha$ reviewers.

Then, we will give the defination of $w, \delta, t$.

### 4.2.1 the Defination of Reviewer Confidence

In general, we trust a reviewer if he or she is able to approximate the real rating of the item in all of his or her scoring. Therefore, we take the Pearson coefficient between the reviewer's score and the item's true rating as the main factor of our model. At the same hand, if the reviewer makes fewer reviews, e. g., only one or two, then we remain skeptical of this decision even if his or her Pearson coefficient is significantly high. Therefore, we will still introduce penalty factors that are used to reduce the reputation scores of those reviewers with a small number of reviews.

$$TR_i = \frac{lg(k_i)}{max\,\{k_j\}} \cdot \frac{1}{k_i} \cdot \sum(\frac{r_{i\alpha} - \overline{r_i}}{\sigma_{r_i}}) \cdot (\frac{Q_\alpha - \overline{Q_i}}{\sigma_{Q_i}})(i \in C_\alpha, j \in C, \alpha \in O)$$

- $k$ denotes the posted review number of the reviewer
- $r_{i\alpha}$ denotes the i reviewer to the good $\alpha$
- $\sigma$ denotes the standard deviation.

Although we obtained Pearson's correlation coefficient for each customer, we cannot yet use it as a confidence level for each customer. In order to make higher reputation values available to those with higher $TR$ values, we need to add the process of reputation redistribution. The main process is defined as

$w_i = TR_i^\theta \frac{\sum TR_j}{\sum TR_j^\theta}$

- $\theta$ is a tunable parameter, which have an influence on reputation redistribution.
- $TR$ denotes the Pearson correlation coefficience

### 4.2.2 the Defination of Content Confidence

For each comment, there is no doubt that its own content should be valued. Whether the commenter is a vine, the emotional intensity of the comment content, the depth of the comment, and the helpfulness of the comment should all be important features to consider in the model. Therefore, we consider using the fuzzy integrated evaluation method to establish the weights of each feature and use TOPSIS to determine the final confidence level of the comment content.

The following table shows the features we will apply.

Table 2: Feature Table

| Sign | Defination | Description |
|---|---|---|
| $H_1$ | $\frac{helpful\_vote}{total\_vote}$ | the ratio of helpness and total votes |
| $H_2$ | None | the degree of emotion expression |
| $H_3$ | $\frac{N_a}{N_t}$ | attribute word ratio |
| $H_4$ | $\frac{N_b}{N_t}$ | proportion of adjectives |
| $H_5$ | $\frac{ln(N_a+N_b)}{ln(N_t)+1}$ | the depth of review |
| $H_6$ | $0 \quad or \quad 1$ | whether is vine |
| $H_7$ | $0 \quad or \quad 1$ | whether is veried customer |

Fuzzy hierarchical analysis is a quantitative analysis method that combines hierarchical analysis with fuzzy comprehensive evaluation. In this paper, FAHP method is used to determine the weights, and its main algorithm steps are as follows. The data for the importance matrix judgment in the fuzzy hierarchical analysis is obtained from the expert scoring, and the specific method is as follows.

Respondents were asked to score two-by-two comparisons according to the important factors affecting consumers' shopping decisions, and each pair of attribute comparison items was scaled by 0.1-0.9, and users were asked to fill out the importance relationship matrix by comparing the importance of different index factors. The importance matrix is filled out by comparing the importance of different index factors. The importance relationship is expressed as an examination function f (x, y), which represents the importance scale of factor x and factor y for the overall, and the importance matrix for f (x, y) is constructed by using the list comparison method. using a list comparison method to construct the priority relation matrix, and the scales are described as shown in Table.

Table 3: Feature Table

| defination | description | measure |
|---|---|---|
| Equally important | x and y are equally important | 0.5 |
| Slightly important | x and y are slightly important | 0.6 |
| Obviously important | x and y are obviously important | 0.7 |
| Special important | x and y are special important | 0.8 |
| Extreme important | x and y are extreme important | 0.9 |

Then we will construct the matrix of determination $C = (c_{ij})_{n\times m}$,

where:

$$C = \begin{matrix} C_{11} & C_{12} & \cdots & C_{1n} \\ C_{21} & C_{22} & \cdots & C_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{n1} & C_{n2} & \cdots & C_{nn} \end{matrix}, (c_{ij} = 0.5)$$

Normalize the elements of the determination matrix.

where:

$$B = (b_{ij})_{m\times n}$$

$$b_{ij} = \frac{a_{ij}}{\sum_{i=1}^{n} a_{ij}}, (i, j = 1, 2, ..., n)$$

Then, the elements in matrix B are summed by rows to obtain vector $C = (c_1, c_2, .., c_n)^T$

where:

$$c_{ij} = \sum_{i=1}^{n} ab_{ij}, (i, j = 1, 2, ..., n)$$

Finally, we would normalize the vector $C$ to obtain the Eigenvector $W = \{w_i, w_2, ..., w_n\}$

In this paper, the improved TOPSIS judging method is used as an online review usefulness ranking filtering model algorithm. The basic idea is: on the basis of determining each. The basic idea is: on the basis of determining the weights of each attribute index, normalizing the original data matrix, calculating the distance between The basic idea is: based on determining the weight of each attribute index, normalizing the original data matrix, calculating the distance between each evaluation object and the optimal and the worst solution, and obtaining the relative the relative proximity of each evaluation object to the optimal solution as the basis for evaluating the superiority and inferiority. The evaluation of each object is based on the normalized data matrix. The specific algorithm steps are as follows. where

(1)In order to eliminate the magnitude effect among different attributes and make each attribute feature equally expressive, the raw data are first normalized. Let the matrix of the multi-attribute decision problem matrix be $A = (a_{ij})_{m\times n}$

$$b_{ij} = \frac{a_{ij} - \overline{a_j}}{s_j}, i = 1, 2, ..., m; j = 1, 2, ...n$$

(2)Create a weighted specification matrix $C_W = (c_{ij}^w)_{m\times n}$

(3)Calculate the distance from each alternative to the positive ideal solution and the negative ideal solution. The distance of the alternative $d_i$ to the positive ideal solution is:

$$s_i^+ = \sqrt{\sum (c_{ij} - c_j^+)^2}, i = 1, 2, ...m$$

The distance of the alternative $d_i$ to the negative ideal solution is:

$$s_i^- = \sqrt{\sum (c_{ij} - c_j^+)^2}, i = 1, 2, ...m$$

(4)Calculate the queuing index value (i.e., the composite evaluation index) for each program:

$$\delta = \frac{s_i^-}{s_i^- + s_i^+}$$

### 4.2.3 the Defination of Timeliness

The timing of the comment publication is also one of the important factors that should be considered as a confidence level. The longer the comment, the less convincing it will undoubtedly be.Also, the curve of timeliness and time is not necessarily linear. Therefore, we intend to introduce an exponential function as a basis function for timeliness.

$$t_i = e^{\Delta t}$$

$\Delta t$ is the difference between the time to read a comment and the time to post it.

### 4.2.4 Iterative method to solve the convergence value

The HITS algorithmwith iterative refinement procedure is a link-basedmodel, which was first used to identify the hubs pages that link to many related authorities in the context of the WWW. As there is a mutually reinforcing relationship between the Hubs and the authorities, the iterative algorithm which can maintain and update numerical weights for each page was introduced to break this circularity. Specifically, the relationship between the hubs and the authorities is described as follows. First, the weights of the hub i and the authority are respectively initialized as $x^{(i)}$ and $y^{(\alpha)}$ after their normalization. Then, the weight $x^{(i)}$ of the hub i is updated by

$$x^{(i)} \leftarrow \sum y^{(\alpha)}$$

where $Y^{(i)}$ denotes the set of the authorities that pointed by the hub i. Conversely, the weight $y^{(\alpha)}$ of the authority is updated by

$$y^{(\alpha)} \rightarrow \sum x^{(i)}$$

This algorithm is borrowed to make $W$ and $Q$ converge eventually by iterating continuously. The combined rating of the product and the reputation of the reviewer are solved[3].

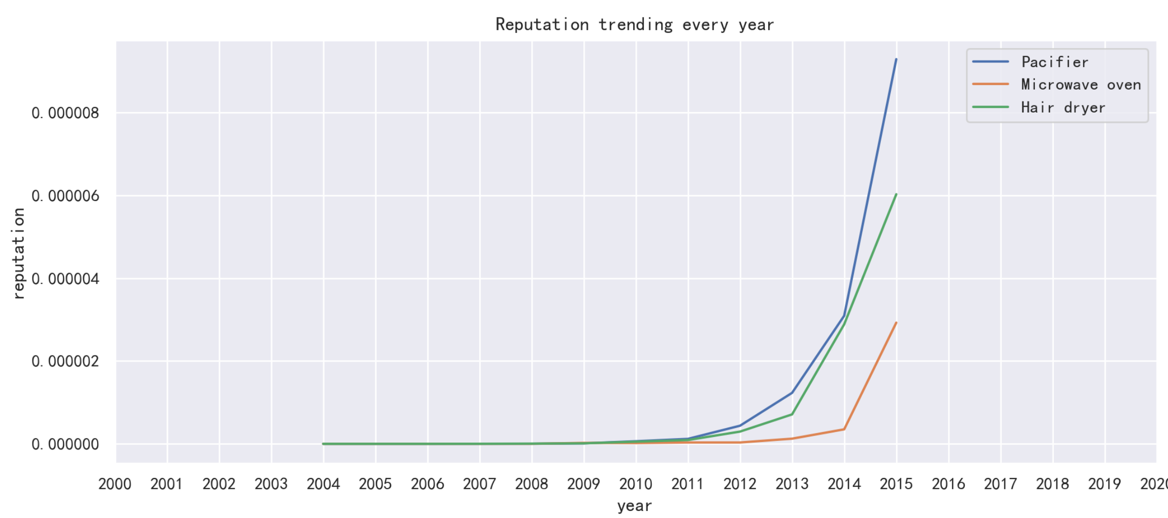


Figure 11: the time-based comprehensive score

The $Q$ calculations show that the three products have broad market prospects and are currently on the rise.

# 5 the Successfulness Prediction Model

Simple Exponential Smoothing (SES) also called Linear Exponential Smoothing, is a model of time series analysis for discrete series with smooth slope or convergence. From the graph, we can notice that the rating gradually converges to a constant. Thus, nothing is better than applying SES to our data to predict the final value converged.

Firstly, we assume that our original series is $y_1, \cdots, y_t$, and a constant $\alpha$

$S_t^{(1)}$ is the predicted value of the series S at moment t

Definition:

$$S_t^{(1)} = \alpha y_t + (1-\alpha) S_{t-1}^{(1)} = S_{t-1}^{(1)} + \alpha (y_t - S_{t-1}^{(1)})$$

i.e.

$$S_t^{(1)} = \alpha \sum_{j=0}^{\infty} (1-\alpha)^j y_{t-j}$$

Eventually we can get the prediction formula

$$\hat{y}_{t+1} = S_t^{(1)} = \alpha y_t + (1-\alpha) \hat{y}_t$$

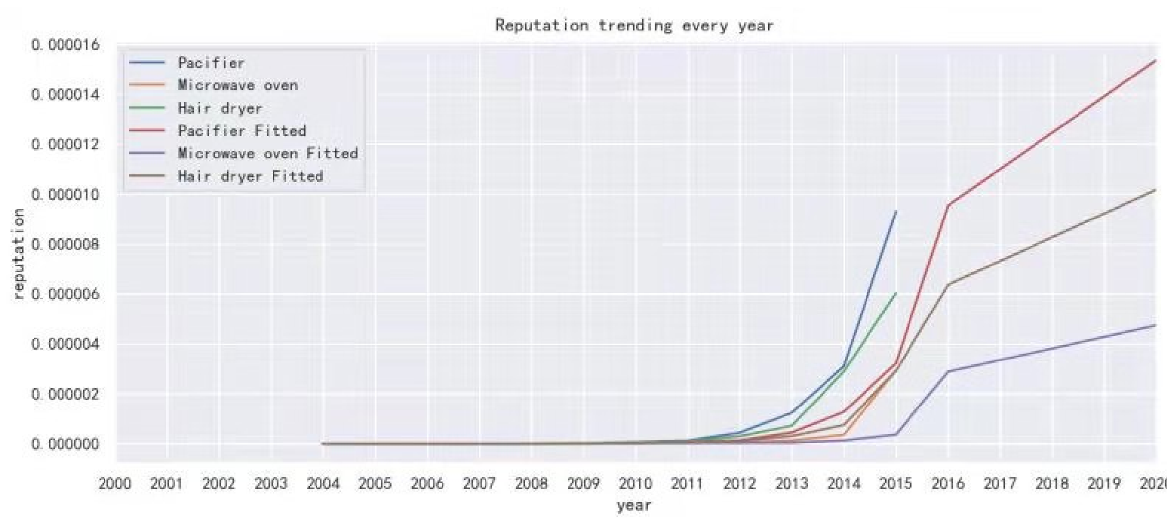


Figure 12: the time-based comprehensive score

The results of the $Q$ projections indicate that three product markets are potentially successful and recommended for investment at some cost.

# 6 Specific Ratings and Descriptors Analysis

In this section, we intend to calculate the correlation between two variables using the Pearson correlation coefficient, whose formula is defined as:

$$R = \sum \left(\frac{r_{i\alpha} - \overline{r_i}}{\sigma_{r_i}}\right) \cdot \left(\frac{Q_\alpha - \overline{Q_i}}{\sigma_{Q_i}}\right)$$

## 6.1 Specific Star Ratings Relevance to Rating Frequency

We divide the timeline appropriately to get enough time points to get the basic data by calculating the average ratings before these points and the sentiment intensity of the comments in the subsequent interval. Bringing in the above formula, the results are obtained as follows:


Figure 13: Pacifier correlation value

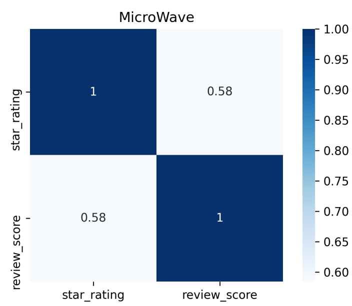


Figure 14: Microwave correlation value


Figure 15: Hair-dryer correlation value

We could see that the value is 0.37, 0.58 and 0.71, which is enough to demonstrate the existence of correlation.

## 6.2 Specific Quality Descriptors Relevance to Rating Levels

We analyzed each comment to obtain their emotional intensity and rating, and again using the formula, we get:

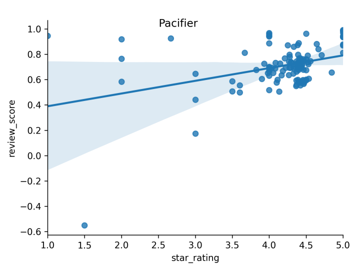


Figure 16: Pacifier correlation

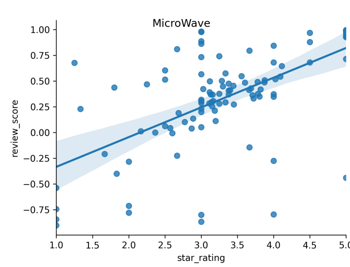


Figure 17: Microwave correlation

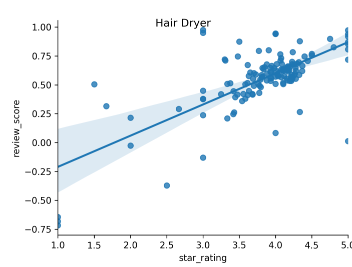


Figure 18: Hair-dryer correlation

With the above figure, it is obvious to obtain that there is a certain correlation between the two variables.

# 7 Sensitivity Analysis

Variance-based sensitivity analysis (often referred to as the Sobol method or Sobol indices, after Ilya M. Sobol) is a form of global sensitivity analysis. Working within a probabilistic framework, it decomposes the variance of the output of the model or system into fractions which can be attributed to inputs or sets of inputs. For example, given a model with two inputs and one output, one might find that 70% of the output variance is caused by the variance in the first input, 20% by the variance in the second, and 10% due to interactions between the two. These percentages are directly interpreted as measures of sensitivity. Variance-based measures of sensitivity are attractive because they measure sensitivity across the whole input space (i.e. it is a global method), they can deal with nonlinear responses, and they can measure the effect of interactions in non-additive systems.

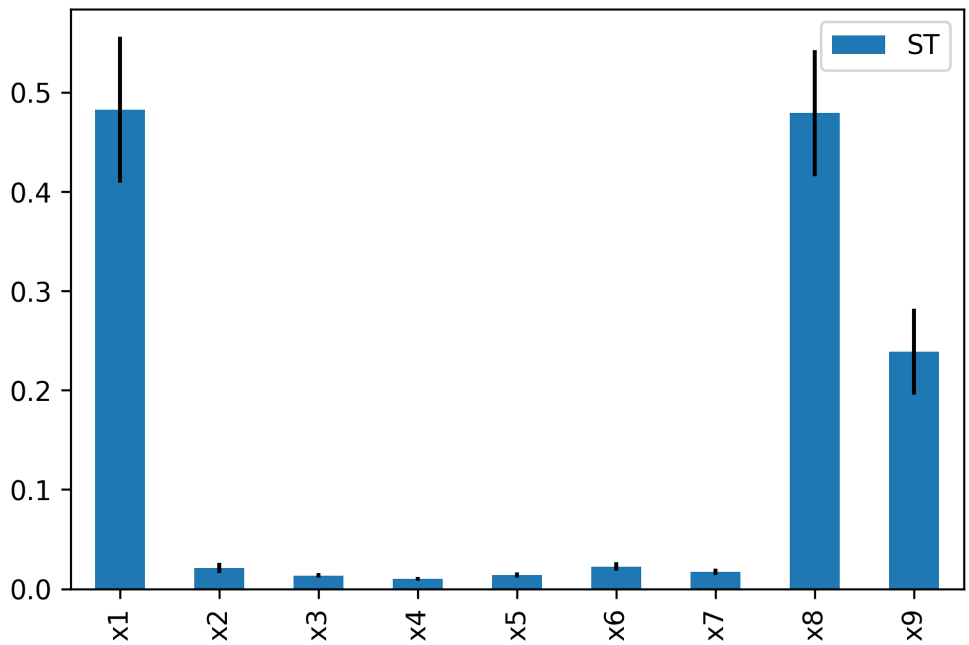


Figure 19: Sensity Analysis

The Figure above shows that reviewer confidence, timeliness, and star rating are the three metrics with high sensitivity. This indicates that our model has high robustness.

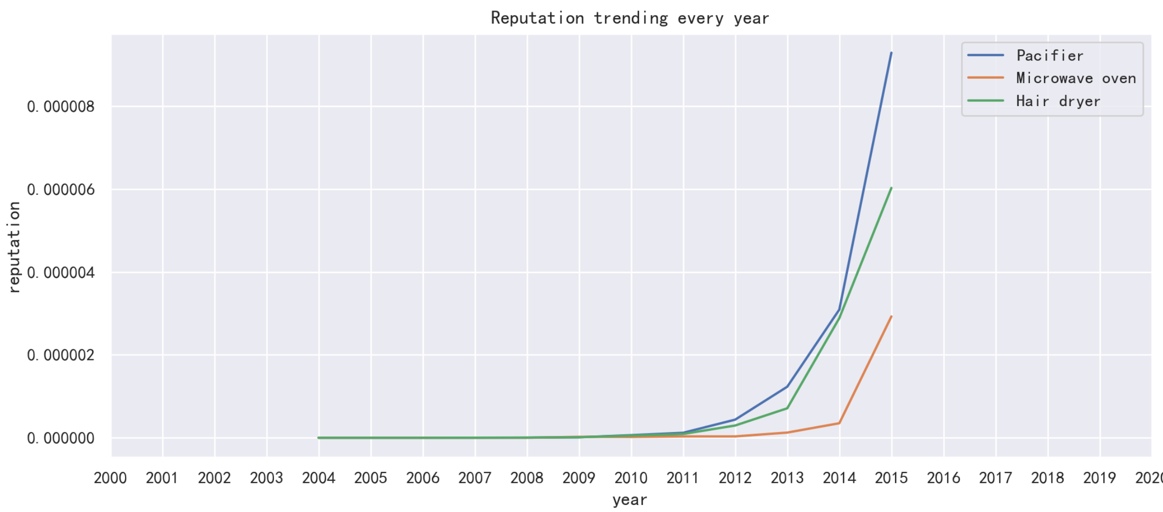


Figure 20: the time-based comprehensive score

# 8 Strengths and Weaknesses

## 8.1 Strengths

- The robustness of the results is enhanced using an iterative approach. We iterate several times to make the combined product score and reviewer reputation approach the convergence value.
- All factors are fully utilized. We constructed several indicators that could reflect product reputation and assigned weights using the FAHP method and constructed confidence levels using TOPSIS. The interpretability of the model is enhanced.
- A reputation redistribution mechanism was introduced. This allows people with high reputation to receive higher weight in each iteration. The risk of malicious comments is circumvented.

## 8.2 Weaknesses

- Underutilization of review text. Only three metrics were defined based on the review text, which were not fully explored.
- It is simpler to use time series to make predictions about the future. Does not have some interpretability.

# 9 Conclusion

In our work, we analyzed the data provided qualitatively and quantitatively to help Sunshine Company better understand the data. We built the Iteration-based Commodity Reputation Redistribution

Model. through this model, Sunshine Company can better track the products, and we also predicted the future trend of the products, and discovered that three markets still have high potential. In addition, we also calculated the correlation coefficient between ratings and review sentiment intensity and found a high correlation.

# 10 Our Letter

Dear Marketing Director of Sunshine Company,

Thank you very much for hiring us as a consultant for this market research. We are going to present our work and researches to you in order to support you and your company to better track the market and formulate strategies.

By analyzing the data you provided us, we found that pacifiers and hair dryers had the majority of 5-star ratings, while microwaves had the majority of 5-star and 1-star ratings. So when your team is ready to enter the microwave oven market, be aware of the polarized word of Mouth.

First, we clean the raw data set and use word search to remove reviews that do not belong to the target product. We defined several measures that can reflect product reputation, analyzed the relationship between average rating, number of reviews, word frequency, and product type, time, rating stars, and visualized them employing line graphs and word cloud graphs.

Based on a binary network of users and products, we mathematically model the confidence of reviews and the overall product rating as objectives. We divide the review confidence into the reviewer's confidence and the confidence of review contents.

For the former, we use the Pearson coefficients between customer ratings and product ratings as the main factors of customer credibility after considering the effects of the instability of a few data ratings and processing them accordingly.

For the latter, we use the external and content features of reviews as variables, determine the weights using the FAHP evaluation method, and then determine the confidence level of review content through the TOPSIS method.

After integrating the review timeliness, we enhanced the robustness through an iterative mechanism of reputation reassignment. Finally, we obtain the final composite rating of the product.

After obtaining the distribution of product reputation scores on the time axis, we will use a time-series approach to fit the reputation curve and make predictions about future reputation scores.

We also conducted a correlation analysis between the sentiment intensity of the reviews and the rating scale of the reviews and found a high correlation coefficient between them. This suggests that the content of users' comments and the ratings they see and type are closely related. Therefore, we recommend that you always spend effort on users' comments. A little carelessness and this will create a positive feedback loop with serious consequences.

If you would like to learn more about our specific research process and results, you can read our paper for more information.

## References

[1] Chien Chin Chen and You-De Tseng. Quality evaluation of product reviews using an information quality framework. *Decision Support Systems*, 50(4):755–768, 2011.

[2] Samita Dhanasobhon, Pei-Yu Chen, Michael Smith, and Pei-yu Chen. An analysis of the differential impact of reviews and reviewers at amazon. com. *ICIS 2007 Proceedings*, page 94, 2007.

[3] Jian Gao and Tao Zhou. Evaluating user reputation in online rating systems via an iterative group-based ranking method. *Physica A: Statistical Mechanics and its Applications*, 473:546–560, 2017.

[4] Hao Liao, An Zeng, Rui Xiao, Zhuo-Ming Ren, Duan-Bing Chen, and Yi-Cheng Zhang. Ranking reputation and quality in online rating systems. *PloS one*, 9(5):e97146, 2014.

# Appendices

Here are simulation programmes we used in our model as follow.

**Input FAHP code:**

```
#!/usr/bin/env python
# coding: utf-8

# In[1]:


import numpy as np
import pandas as pd


# In[73]:


A1 = [0, 0, 0, 0, 0, 0]  #
R1 = [0, 0, 0, 0, 0, 0]  #
W = [0, 0, 0, 0, 0, 0]
R = np.zeros((6, 6), dtype=np.float)
SUM = 0
global N
N = 6

#
A = np.array([
    [0.50, 0.75, 0.80, 0.60, 0.50, 0.55],
    [0.25, 0.50, 0.65, 0.50, 0.30, 0.40],
    [0.20, 0.35, 0.50, 0.40, 0.30, 0.40],
```

```
    [0.40, 0.50, 0.60, 0.50, 0.30, 0.40],
    [0.50, 0.70, 0.70, 0.70, 0.50, 0.75],
    [0.45, 0.60, 0.60, 0.60, 0.25, 0.50]
])


# In[74]:


for i in range(N):
    A1[i] = 0  #
    for j in range(N):
        A1[i] += A[i][j]


# In[75]:


print(A1)


# In[76]:


#
for i in range(N):
    for j in range(N):
        R[i][j] = (A1[i] - A1[j]) / (2 * N) + 0.5


# In[77]:


#
for i in range(N):
    R1[i] = 1
    for j in range(N):
        R1[i] *= R[i][j]
    W[i] = pow(R1[i], 0.2)
    SUM += W[i]


# In[78]:


for i in range(N):
    W[i] = W[i] / SUM


# In[79]:


print(W)
```

## Input TOPSIS code:

```
#!/usr/bin/env python
# coding: utf-8

# In[ ]:


import pandas as pd
import numpy as np


# In[ ]:


Pcf_df = pd.read_csv("./CacheData/delta/Pcf_delta.csv")

Mcw_df = pd.read_csv("./CacheData/delta/Mcw_delta.csv")

Hdr_df = pd.read_csv("./CacheData/delta/Hdr_delta.csv")


# In[ ]:


def topsis(data, weight=None):
    """

    TOPSIS algorithm

    Args:
        data: Features
        weight:

    Returns:
        Result:
        Z:
        weight:

    """
    data = data / np.sqrt((data ** 2).sum())  # normalized

    Z = pd.DataFrame([data.min(), data.max()], index=['', ''])  # best and worst solution

    weight = entropy_weight(data) if weight is None else np.array(weight)  # distance
    Result = data.copy()
    Result[''] = np.sqrt(((data - Z.loc['']) ** 2 * weight).sum(axis=1))
    Result[''] = np.sqrt(((data - Z.loc['']) ** 2 * weight).sum(axis=1))

    # composite score index
    Result[''] = Result[''] / (Result[''] + Result[''])
    Result[''] = Result.rank(ascending=False)['']

    return Result, Z, weight
```

```
# In[ ]:


Pcf_res = topsis(Pcf_df, [0.1903, 0.1531, 0.1383, 0.1564, 0.1954, 0.1665])

Mcw_res = topsis(Mcw_df, [0.1903, 0.1531, 0.1383, 0.1564, 0.1954, 0.1665])

Hdr_res = topsis(Hdr_df, [0.1903, 0.1531, 0.1383, 0.1564, 0.1954, 0.1665])


# In[ ]:


Pcf_res[0]
```

**Input TFIDF code:**

```
"""
-------------------------------------
# -*- coding: utf-8 -*-
# @Time    : 2021/1/25 16:13:42
# @Author  : Giyn
# @Email   : giyn.jy@gmail.com
# @File    : TF_IDF.py
# @Software: PyCharm
-------------------------------------
"""

import nltk
import math
import string

from nltk.corpus import stopwords
from collections import Counter
from nltk.stem.porter import *


def get_tokens(text: str):
    """

    participle

    Args:
        text: text

    Returns:
        tokens

    """
    lower = text.lower()
    remove_punctuation_map = dict((ord(char), None) for char in string.punctuation)
    no_punctuation = lower.translate(remove_punctuation_map)
```

```python
    tokens = nltk.word_tokenize(no_punctuation)

    return tokens


def stem_tokens(tokens, stemmer):
    """

    stemming

    Args:
        tokens: tokens
        stemmer: stemmer

    Returns:
        stemmed

    """
    stemmed = []
    for item in tokens:
        stemmed.append(stemmer.stem(item))

    return stemmed


def tf(word, count):
    return count[word] / sum(count.values())


def n_containing(word, count_list):
    return sum(1 for count in count_list if word in count)


def idf(word, count_list):
    return math.log(len(count_list)) / (1 + n_containing(word, count_list))


def tfidf(word, count, count_list):
    return tf(word, count) * idf(word, count_list)


def count_term(text):
    tokens = get_tokens(text)
    filtered = [w for w in tokens if not w in stopwords.words('english')]
    stemmer = PorterStemmer()
    stemmed = stem_tokens(filtered, stemmer)
    count = Counter(stemmed)
    return count


def tf_idf(texts: list):
    countlist = []

    for text in texts:
```

```
        countlist.append(count_term(text))
    tf_idf_dict = {}
    for i, count in enumerate(countlist):
        # print("Top words in document {}".format(i + 1))
        scores = {word: tfidf(word, count, countlist) for word in count}
        sorted_words = sorted(scores.items(), key=lambda x: x[1], reverse=True)
        for word, score in sorted_words[:5]:
            tf_idf_dict[word] = round(score, 5)
            # print("\tWord: {}, TF-IDF: {}".format(word, round(score, 5)))

        yield tf_idf_dict


if __name__ == "__main__":
    texts = ['Giyn likes guitar.',
             'Helen likes guitar too.',
             'Giyn also likes piano.']
    for i in tf_idf(texts):
        print(i)
```