**Natural Language Processing -**

**Amazon Fine Food Reviews**OPIM 5503 Data Analytics Using R

Team Name: We R Ready

Team Members: Jiehao Pan, Santosh Mysore Vasudevan,

Siqi Wan, Yan Tan, Yanjun Yang

**Contents**

1. Introduction

2. What is Text Mining

3. Dataset Description

4. Data Preparation

5. Sentiment Analysis

6. Topic Modeling

7. Recommendation System

8. Package Used

References

**1. Introduction**

With the development of Big Data Analytics, Natural Language Processing (NLP) has been deep into our life. As a part of artificial intelligence, it can make machines to correctly read and understand the language human speak. The goal of natural language processing is to allow interactions so that non-programmers can gain some useful information from the computing systems. So in a way, NLP makes our life easier.

In this project, we will use the data of Amazon Food Reviews to do text mining by finishing text preprocessing, doing sentiment analysis to divide the reviews into two aspects: positive and negative. Then we will complete topic modeling to list several most frequent topics in both positive reviews and negative reviews. Finally, we are going to give somen recommendations to customers of Amazon by using recommendation system.

**2. What is Text Mining**

Text mining, also referred to as text data mining, roughly equivalent to [text analytics](https://en.wikipedia.org/wiki/Text_mining#Text_mining_and_text_analytics), is the process of deriving high-quality [information](https://en.wikipedia.org/wiki/Information) from[text](https://en.wikipedia.org/wiki/Plain_text). High-quality information is typically derived through the devising of patterns and trends through means such as [statistical pattern learning](https://en.wikipedia.org/wiki/Pattern_recognition). Text mining usually involves the process of structuring the input text, deriving patterns within the [structured data](https://en.wikipedia.org/wiki/Structured_data), and finally evaluation and interpretation of the output. 'High quality' in text mining usually refers to some combination of [relevance](https://en.wikipedia.org/wiki/Relevance_(information_retrieval)), [novelty](https://en.wikipedia.org/wiki/Novelty_(patent)), and interestingness. Typical text mining tasks include[text categorization](https://en.wikipedia.org/wiki/Text_categorization), [text clustering](https://en.wikipedia.org/wiki/Text_clustering), [concept/entity extraction](https://en.wikipedia.org/wiki/Concept_mining), production of granular taxonomies, [sentiment analysis](https://en.wikipedia.org/wiki/Sentiment_analysis), [document summarization](https://en.wikipedia.org/wiki/Document_summarization), and entity relation modeling (i.e., learning relations between [named entities](https://en.wikipedia.org/wiki/Named_entity_recognition)).

Essentially, the overarching goal of text mining is to turn text into data for analysis, via application of [natural language processing](https://en.wikipedia.org/wiki/Natural_language_processing) (NLP) and analytical methods.

Sentiment analysis refers to the use of [natural language processing](https://en.wikipedia.org/wiki/Natural_language_processing), [text analysis](https://en.wikipedia.org/wiki/Text_analytics) and [computational linguistics](https://en.wikipedia.org/wiki/Computational_linguistics) to identify and extract subjective information in source materials. Sentiment analysis is widely applied to reviews and social media for a variety of applications, ranging from [marketing](https://en.wikipedia.org/wiki/Marketing) to [customer service](https://en.wikipedia.org/wiki/Customer_relationship_management). Generally speaking, sentiment analysis aims to determine the attitude of a speaker or a writer with respect to some topic or the overall contextual polarity of a document.

Recommender systems typically produce a list of recommendations in one of two ways – through [collaborative](https://en.wikipedia.org/wiki/Collaborative_filtering) and [content-based filtering](https://en.wikipedia.org/wiki/Content-based_filtering) or the personality-based approach.

**3. Dataset Description**

Our raw data has 568,454 observations of 10 variables.

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Type** | **Description** |
| Id | Int | Id |
| ProductId | Factor | Unique identifier for the product |
| UserId | Factor | Unique identifier for the user |
| HelpfulnessNumerator | Int | Number of users who found the review helpful |
| HelpfulnessDenominator | Int | Number of users who indicated whether they found the review helpful |
| Score | Int | Rating between 1 and 5 |
| Time | Int | Timestamp for the review |
| Summary | Factor | Brief summary of the review |
| Text | Factor | Text of the review |

**4. Data Preparation**

**4.1 Data Preparation of Sentiment Analysis**

Firstly, in view of the huge raw data, we decided to extract a part of them to do text mining.

For sentiment analysis, we extract 20% of the raw data as a sample.



Secondly, what we are interested in is the words refer to positive attitude or negative attitude. So we remove the observations with neutral sentiment (Score = 3).



Thirdly, we split the data set into 60% train and 40% test.



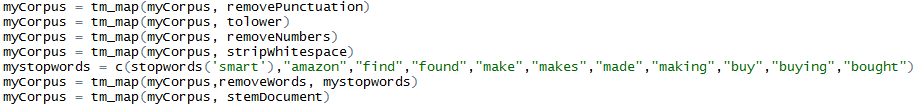
Then we add a new column showed sentiment (positive or negative) of the reviews in both train data and test data .



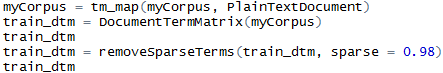
After that, we extract the ‘reviews’ column as text data for transformation and convert the text data to corpus.



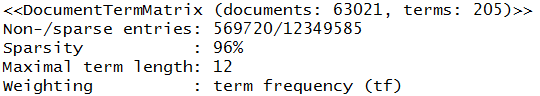
Afterwards, we did some basic text preprocessing like transforming all characters to lowercase, removing punctuations,numbers and white spaces, removing stop words by using stoplist ‘smart’ in rm package and some other customized stop words like displayed below. Then applying the stemming.



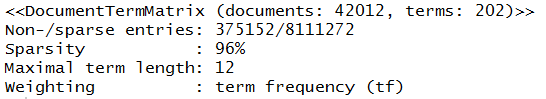
Finally, we convert the corpus to a plain text document, create the Document Term Matrix based on the corpus and reduce the sparsity of dtm to 98%.



The train\_dtm is as follow:



For test data, we repeated the above process to dispose it and got the test\_dtm as below:

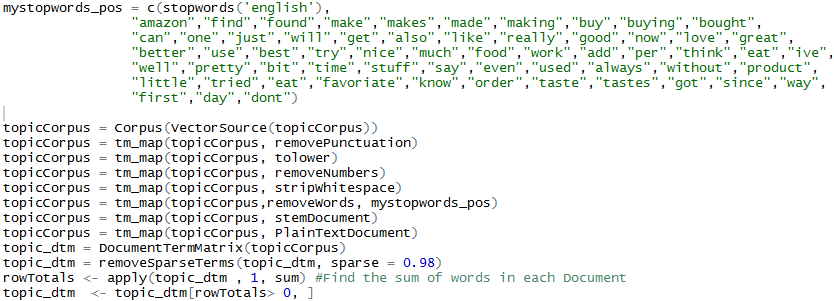


**4.2 Data Preparation of Topic Modeling**

In this part of text mining, we choose to extract 5% of raw data which is different from that in sentiment analysis and then extracting the ‘reviews’ column as text data.



Except that, the preparation for topic modeling is much similar to that of sentiment analysis. Such as converting text data to corpus, removing punctuations, numbers, whitespace and stopwords, applying stemming, converting corpus to a plain text document, creating the Document Term Matrix and reduce the sparsity of it to 0.98. However, ‘my stop words’ we chose are more specific than that of sentiment analysis.



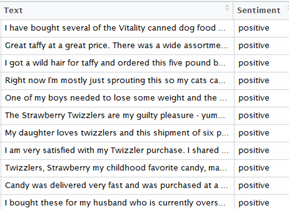
Besides, we found the sum of words in each document.



**5. Sentiment Analysis**

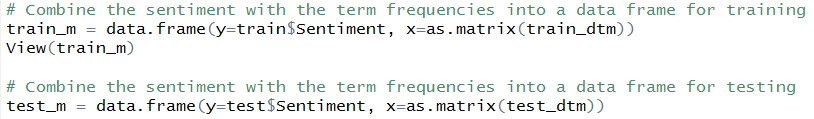
Sentiment Analysis (SA), also known as opinion mining, is used to identify and classify the sentiments of opinions or reviews in a piece of text regarding to the topic and the overall contextual polarity of a document. These sentiments may be divided into positive, neutral and negative.

In sentiment analysis, the most important part is to classify the polarity of a given text in the document. In our case -Amazon find foods reviews, we have a range of score which ranks the sentiment from very negative (score = 1) to very positive (score = 5). We filled the neutral sentiment (score = 3) out before analysis. Then we added a column called “sentiment” which contains “positive” (score>3) and “negative” (score<3).



**5.1 Sentiment Classification**

As we mentioned in data preparation before, we divide the dataset into train (60%) and test (40%). Then we combine the sentiment with the term frequencies for train and test as follow.



Train\_m and Test\_m for modeling are as follow:

Train\_m: 63098 observations of 204 variables

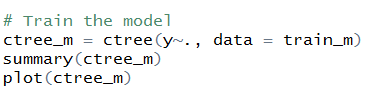
Test\_m: 42064 observations of 201 variables

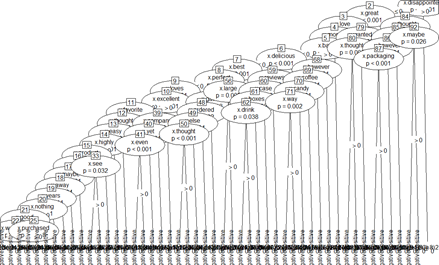
After we researched on several classification methods, we decided to use the following three classification methods to build the model: **ctree classifier**, **Naive Bayes classifier** and **ksvm classifier**.

**5.1.1 Ctree Classifier**

CTree classification (conditional inference trees) is a basic decision tree model with extra information in the terminal nodes. It is a main function in PARTY package.

The code of ctree classification is as below:

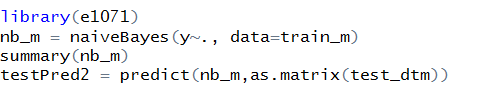




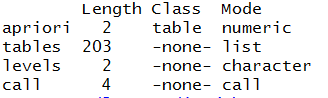
**5.1.2 Naive Bayes Classifier**

The Naive Bayes classifier is a simple probabilistic classifier which is based on Bayes theorem with strong and naive independence assumptions to predict binary classification model. It is one of the most basic text classification techniques with various applications in email spam detection, document categorization, language detection, sentiment detection and automatic medical diagnosis. It is one of the most basic text classification techniques used in various applications.

In this case, we used this classification method to train our model as the second method. The code is as below:

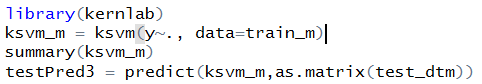


The summary of naiveBayes is as below:



**5.1.3 KSVM Classifier**

KSVM (Kernel Support Vector Machines) are a tool for classification and regression. This function is in the package “kernlab”. The code is as below:

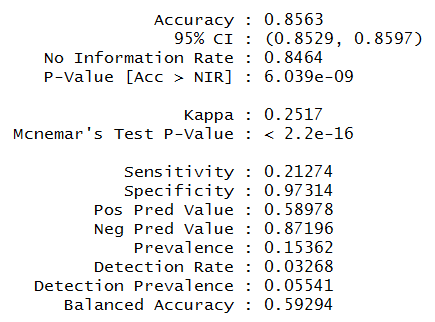


**5.2 Model Comparison**

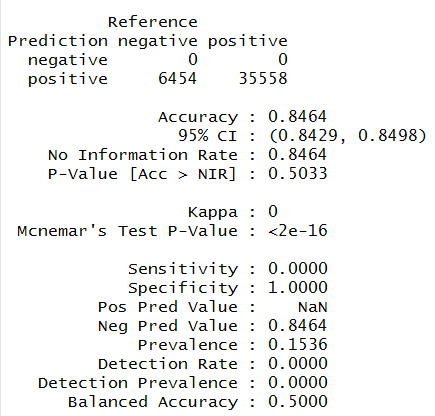
We can assess the quality of the model by constructing a [confusion matrix](http://en.wikipedia.org/wiki/Confusion_matrix).

The results of these three models are as below:

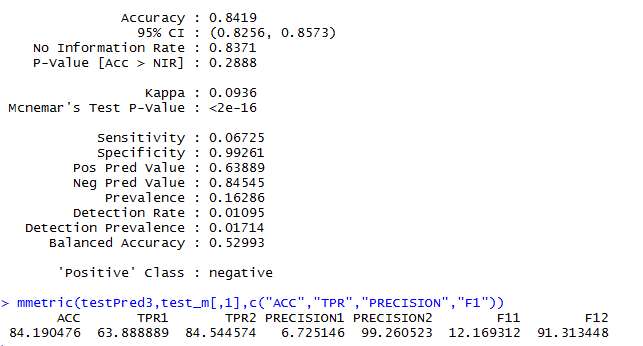
CTree Classifier:



naive Bayes Classifier:



KSVM Classifier:



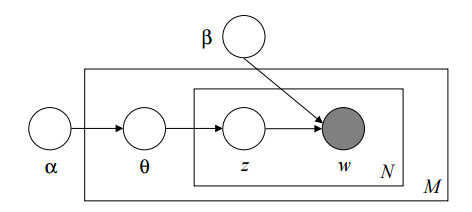
The prediction accuracy of this classification model is given by the proportion of the total number of correct predictions. The accuracy for this model turns out to be around 84% and the first method - ctree classifier showed the best result with 85% accuracy rate.

**6. Topic Modeling**

Topic model is a type of [statistical model](https://en.wikipedia.org/wiki/Statistical_model) for discovering the abstract "topics" that occur in a collection of documents(wikipedia). The most frequently used algorithm for topic modeling is Latent Dirichlet Allocation algorithm.

**6.1 Latent Dirichlet Allocation**

The key insight into LDA is the premise that words contain strong semantic information about a document. Therefore, it is reasonable to assume that documents on roughly similar topics will use the same group of words.   
The graphical model representation of LDA is the following:



The boxes are “plates” representing replicates.The outer plate represents documents, while the inner plate represents the repeated choice of topics and words within a document.

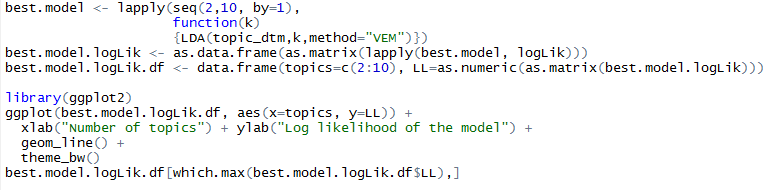
**6.2 Topic Model**

There are three steps for topic modeling work: text preparation, optimize number of topics and topic modeling. As it is mentioned in part 4.2, the text preparation generate a corpus for topic model. The following part will talk about optimize number of topics and final topic model.

6.2.1 Optimize number of topics

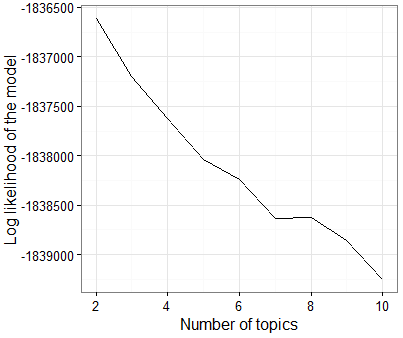
Finding the number of topics which can optimize the log likelihood of topic model is a crucial part of topic modeling work.

We use the following function to find the optimize number of topics:

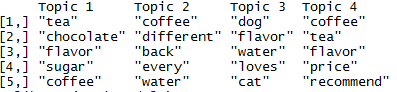


We plot the log likelihood vs number of topics(2-10 topics) as follows:

From the plot we can conclude the model performs better when number of topics are less. This situation happens because our data is food review text with similar one topic.



Under this situation, we pick 4 topics as our final model and getting the following result:



* Tea, coffee and pet food are products customers purchase most
* Customer reviews often focus on food flavor and price
* The reviews can be separated into 4 following topics:
  + Topic 1: product flavor
  + Topic 2: coffee related (different coffee)
  + Topic 3: pet food related (food flavor)
  + Topic 4: product price

In general, topic modeling has wild usage in Natural Language Processing area. It helps us extract and understand the general topic of certain text. It can be a powerful tool in analyzing text information in website, articles and social media platform.

**7. Recommendation System**

There is an extensive class of Web applications that involve predicting user responses to options. Such a facility is called a Recommendation system.

In this project, we felt the need to extend the scope beyond text mining and topic selection as we wanted to find ways to retain customers, which could be achieved by providing accurate recommendations about different products to the customer.

Any recommendation is built based on the utility matrix as given below which contain the ratings. The rows denote different users and the columns denotes the items they have rated.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Item 1 | Item 2 | Item 3 | Item 4 | Item 5 | Item 6 | Item 7 | Item 8 |
| User A | 4 |  |  | 5 | 1 |  |  | 1 |
| User B | 5 | 5 | 4 |  |  |  |  |  |
| User C |  |  |  | 2 | 4 | 5 |  |  |
| User D |  | 3 |  |  |  |  | 3 |  |

Recommendation systems can be implemented by many methods. They are described as follows:

**7.1 Content based recommender system:** This method examines the properties of the item recommended. For instance, if an Amazon prime user watches content related to cars and other automobile, he will be recommended to watch content classified under the genre “cars and automobile”.

**7.2 Collaborative filtering techniques:**

**7.2.1 User based collaborative filtering:**  This method works on the basis of user similarity. It finds the set of users who rate similar to say User A, and recommend items to User A which have not been rated by User A but has been rated by other users who rate similar to User A.

The similarity is calculated by Cosine similarity :

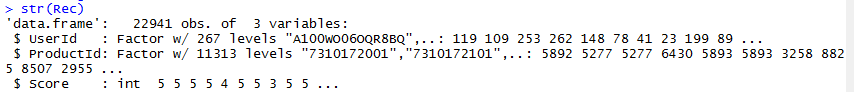
sim(A, B) = cos(rA, rB) = rA.rB/||A||.||B||

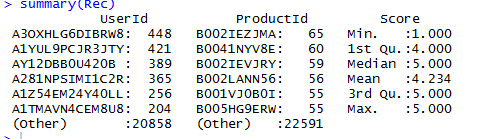
**7.2.2 Item based collaborative filtering:** This method for item i, finds other similar items and estimates rating for item i based on ratings for similar items. Similar similarity metrics can be used as in user based collaborative filtering.

**7.3 recommenderlab:**

The R package used for building the recommender system was recommenderlab.

The dataset considered has the following structure:



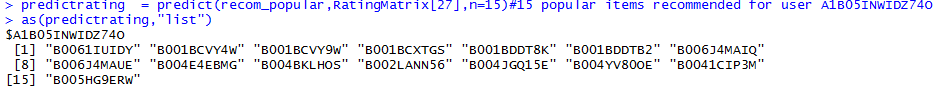


Before we build the recommendation system, we need to convert the dataset into realRatingMatrix format. Another challenge is the sparse matrix, as there are a lot more products when compared to the number of users. To deal with this issue only users who have at least reviewed 20 items have been considered.



We use three different methods to build recommendation system for user M. A. Ramos with UserId A1B05INWIDZ74O.

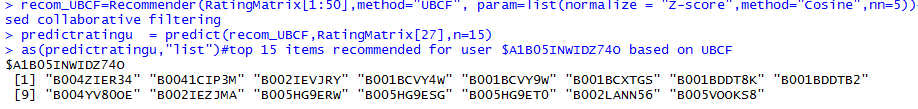
1. method = "POPULAR": This method gives a list of the top 15 popular recommended items.



|  |  |  |
| --- | --- | --- |
| Sl no. | ProductId | Description |
| 1 | B0061IUIDY | Higgins & Burke, Black Tea |
| 2 | B001BCVY4W | Petite Cuisine Sesame Chicken Entree for Cats |
| 3 | B001BCVY9W | Petite Cuisine New England Crab Cake for Cats |
| 4 | B001BCXTGS | Petite Cuisine Red Snapper Entree for Cats |
| 5 | B001BDDT8K | Petite Cuisine Yellow Fin Tuna Entree for Cats |
| 6 | B001BDDTB2 | Petite Cuisine Variety Pack (Yellowfin, Snapper, Tuna & Sole, Tuna & Shrimp) for Cats |
| 7 | B006J4MAIQ | Nature's Path Love Crunch Premium Organic Granola |
| 8 | B006J4MAUE | Nature's Path Love Crunch Premium Organic Granola |
| 9 | B004E4EBMG | MIO Mango Peach |
| 10 | B004BKLHOS | Back To Nature Golden Honey Oat Grahams |
| 11 | B002LANN56 | Chef Michael's Grilled Sirloin Dry Dog |
| 12 | B004JGQ15E | Snackwell's White Fudge Drizzle Caramel Popcorn |
| 13 | B004YV80OE | Kraft Velveeta Chicken and Broccoli Skillets Dinner Kit |
| 14 | B0041CIP3M | Prima Taste Rendang Curry Sauce Kit |
| 15 | B005HG9ERW | Essentia 9.5 pH Drinking Water |

2. method= “UBCF”

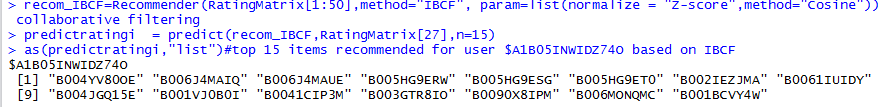
We used User based collaborative filtering to recommend top 15 products.. In UBCF, 5 neighbours were considered to calculate user similarity. Cosine measure was used.



|  |  |  |
| --- | --- | --- |
| Sl no. | ProductId | Description |
| 1 | B004ZIER34 | Puroast Low Acid Coffee French Roast |
| 2 | B0041CIP3M | Prima Taste Rendang Curry Sauce Kit |
| 3 | B002IEVJRY | illy issimo Coffee Drink, Cappuccino |
| 4 | B001BCVY4W | Petite Cuisine Sesame Chicken Entree for Cats |
| 5 | B001BCVY9W | Petite Cuisine New England Crab Cake for Cats |
| 6 | B001BCXTGS | Petite Cuisine Red Snapper Entree for Cats |
| 7 | B001BDDT8K | Petite Cuisine Yellow Fin Tuna Entree for Cats |
| 8 | B001BDDTB2 | Petite Cuisine Variety Pack (Yellowfin, Snapper, Tuna & Sole, Tuna & Shrimp) for Cats |
| 9 | B004YV80OE | Kraft Velveeta Chicken and Broccoli Skillets Dinner Kit |
| 10 | B002IEZJMA | illy issimo Coffee Drink, Caffè |
| 11 | B005HG9ERW | Essentia 9.5 pH Drinking Water,20-oz. |
| 12 | B005HG9ESG | Essentia 9.5 pH Drinking Water |
| 13 | B005HG9ET0 | Essentia 9.5 pH Drinking Water, 1.5 Liter |
| 14 | B002LANN56 | Chef Michael's Grilled Sirloin Dry Dog Food |
| 15 | B005VOOKS8 | Marley Coffee & Tea Mountain Roast Swiss Water, Decaf |

3. method= “IBCF”

We used Item based collaborative filtering to recommend top 15 products.Cosine measure was used.



|  |  |  |
| --- | --- | --- |
| Sl no. | ProductId | Description |
| 1 | B004YV80OE | Kraft Velveeta Chicken and Broccoli Skillets Dinner Kit |
| 2 | B006J4MAIQ | Nature's Path Love Crunch Premium Organic Granola |
| 3 | B006J4MAUE | Nature's Path Love Crunch Premium Organic Granola, Aloha Blend |
| 4 | B005HG9ERW | Essentia 9.5 pH Drinking Water, 20-oz. |
| 5 | B005HG9ESG | Essentia 9.5 pH Drinking Water, 1 Liter |
| 6 | B005HG9ET0 | Essentia 9.5 pH Drinking Water, 1.5 Liter |
| 7 | B002IEZJMA | illy issimo Coffee Drink, Caffè |
| 8 | B0061IUIDY | Higgins & Burke, Black Tea, Earl Grey |
| 9 | B004JGQ15E | Snackwell's White Fudge Drizzle Caramel Popcorn |
| 10 | B001VJ0B0I | Purina Beneful Originals With Real Beef |
| 11 | B0041CIP3M | Prima Taste Rendang Curry Sauce Kit, 12.7-Ounce Boxes |
| 12 | B003GTR8IO | Starbucks Natural Fusions Vanilla Ground Coffee |
| 13 | B0090X8IPM | Starbucks Natural Fusions Vanilla Ground Coffee, 11 Ounce |
| 14 | B006MONQMC | Vitamin Squeeze Energy Drink, Fruit Punch |
| 15 | B001BCVY4W | Petite Cuisine Sesame Chicken Entree for Cats |

**8. R Package Used**

**tm:** Used for sentiment analysis. Processes text sentences and create a corpus.

Functions: Corpus(), tm\_map(), DocumentTermMatrix()

Other packages in SA: party, caret, e1071, kernlab

**topicmodels:** Provides an interface to the C code for Latent Dirichlet Allocation (LDA) models.

Functions: LDA()

**recommenderlab:**Provides a research infrastructure to test and develop recommender algorithms including UBCF, IBCF, FunkSVD and association rule-based algorithms.

**References**

[1] Wikipedia, Text Mining

[2] Wikipedia, Sentiment Analysis

[3] Wikipedia, Latent Dirichlet Allocation

[4] Yanchang Zhao. (2012). *R and Data Mining: Examples and Case Studies*, Elsevier.

[5] Mining of Massive Datasets. Jure Leskovec,Anand Rajaraman and Jeffrey D. Ullman