Sentiment Analysis- Strategy for Text Pre-Processing

Bhumika Pahwa
Research Scholar
Banasthali Vidyapith,
Jaipur, Rajasthan

S. Taruna,PhD
Associate Professor
JK Lakshmipat University,
Jaipur, Rajasthan

Neeti Kasliwal,PhD
Associate Professor
IIHMR University,
Jaipur, Rajasthan.

ABSTRACT
It is taxing to understand the current trends in the online market and then abridge the general opinions about the products due to the existence of diversified social media data. This has created a need for real time opinion mining which is analysis of the sentiments that classifies the text into positive and negative emotion polarities. In this paper, the author explores the most important step in sentiment analysis that is data pre-processing and analyses the different techniques used for pre-processing in R. The results show that using library packages provides better results with respect to the method where direct functions are used.

Keywords
Sentiment analysis, data pre-processing, Tm Library, Natural language processing.

1. INTRODUCTION
Sentiment analysis of reviews is the process of investigating the online reviews for determining the overall feeling or opinion about the product. A large number of reviews on the internet showcase the current form of user's feedback and it becomes hard to find out the latest trends and sum up general opinions due to the diversity and size of social media data which creates the dire need of real time opinion extraction and mining. Settling on the sentiment of opinion is a challenging issue because of the subjective nature of reviews [1].

Sentiment analysis is basically a classification task that classifies the orientation of a text into positive or negative or neutral and machine learning is one of the most widely used approaches towards sentiment classification [2].

The process of sentiment analysis is discussed in figure 1, which consists of collecting customer reviews and the pre-processing of data, then aspect identification followed by sentiment classification and aspect ranking and overall ranking of the sentiment.

Each step in figure 1 shows data in a particular state, the First state shows the raw data that is just how the data has been collected and it may contain wrong data types or labels, special characters, white spaces and different font sizes. This data cannot be directly fed into the model for analysis without any pre-processing. After this pre-processing, the data becomes technically correct, which means that now data can be read into the model without errors.

Text Pre-processing is the process of filling in the missing values, smoothen the noisy data and removing outliers and resolving inconsistencies from the data collected from primary or secondary sources for analysis or model building.

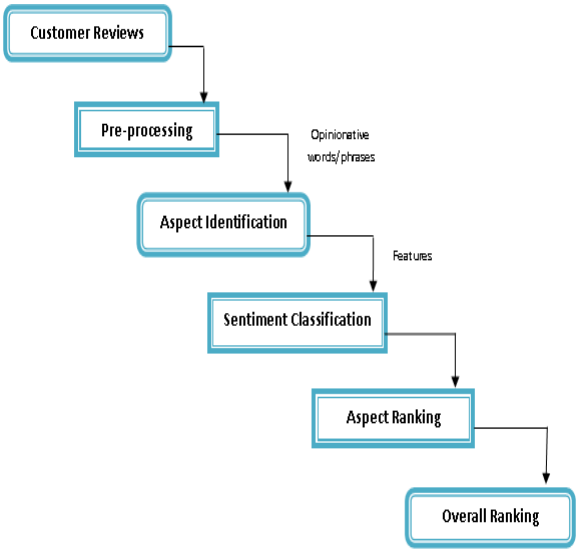

Figure 1: Sentiment analysis process

Pre-processing the data is the process of cleaning and preparing the text for the classification process. The necessity for this step lies in the fact that online texts usually contain noise and uninformative parts such as HTML tags, scripts and advertisements. In addition, on words level, many words in the text do not have an impact on the general orientation of text. Since each word in the text is treated as one dimension, keeping irrelevant words increases the dimensionality of the problem and hence makes the classification more difficult [3]. These difficulties not only manifest themselves in the robustness of the analysis, but also in the computational complexity of the classification process [4].

Text preprocessing is not just an imperative part of data science process, but it is also the most time taking part.

This paper focuses on the methods used for text pre-processing and analyses them to understand which method is more beneficial for sentiment analysis while working on R.

1.1 Importance of Text Pre-Processing
Converting the text into something that can be absorbed by the tool is a very intricate process and has four different parts.

- Cleaning: It consists of removing the less useful parts of text through stop word removal, dealing with capitalization and characters.
- Annotation : It consists of the applying a scheme to the text and includes part-of-speech tagging
- Normalization: Consists of mapping the terms in a scheme.
- Analysis: Consists of manipulating the dataset for feature analysis.

2. BACKGROUND

After data collection process, the second step is to remove the noise from the data and convert the data into a format that is acceptable by the model to be used and this is done by first removing the entries with missing values, removing the special characters if the data is in textual form, converting the text into lower case since people make use of capital words while emphasizing on a word and this creates dissimilarity in the data, therefore all the data needs to be converted into lower case, then stop-words are removed from the textual data, like 'is', 'and', 'the' etc., removing punctuation marks and removal of white spaces since punctuation symbols and white spaces don't have any meanings.

Mong Li Lee [5] in his paper examined the problem of detecting and removing duplicates records. Several different techniques to pre-process the records before sorting them so that potentially matching records will be brought to close neighborhood subsequently.

Rahm, Hong Hai Do [6] in their paper defined the various data cleaning problems and current approaches like Single source problems and Multisource problems and Data quality problems.

Hamid Ibrahim Housien et al[7] in their paper study the data scrubbing algorithms and frameworks in data warehouse.

Nidhi Choudhary[8] in his paper study the various problems and approaches in Data cleaning.

Joseph M. Hellerstein[9] in his paper discuss the quantitative cleaning of large databases**,** and defines the approaches to improve data. quality.

Rajashree Y.Patil et al [10] have discussed various data cleaning algorithms for data warehouse.

Heiko Müller et al[11] in their paper discussed the various data cleaning process and compare the data cleaning frameworks.

Mong Li Lee et al [12] in their research paper they proposed a generic knowledge based framework for effective data cleaning that implements existing cleaning strategies and more.

Kofi Adu-Manu Sarpong et. al[13] in their paper conceptualized the data cleansing process from data acquisition to data maintenance. Data Cleansing is an activity involving a process of detecting and correcting the errors and inconsistencies in data warehouse.

Taoxin Peng [14] in his paper presented a framework for How to improve the efficiency while performing data cleaning and How to improve the degree of automation when performing data cleaning, which provides an approach to managing data cleaning in data warehouse by focusing on the use of data quality dimensions, and decoupling a cleaning process into several sub-processes.

This paper studies the methods used for cleaning the data collected in the form of customer reviews from various e-commerce websites. Working in R, the author first used the traditional methods provided by Natural language processing functions, pre-defined in R and reduced the noise by a considerable amount. Initially, the data was in the form of a table(csv) and can be directly used for cleaning with the help of functions such as gsub(), tolower() etc.

Second method used for text pre-processing involves the use of library packages such as Tmlib, but data cannot be used in the table format, hence a corpus is created which is then used for text pre-processing.

The next section discusses the two methods in detail.

3. METHODOLOGY AND ANALYSIS

The traditional method involves the use of predefined functions in R for cleaning the inconsistencies from the review dataset collected from various e-commerce websites. This method requires using the dataset directly in the tabular form. For this, the dataset in .csv format is directly imported in R and various functions are used for removing the punctuation marks, converting the data into lowercase and then writing all the content in the file as shown in figure 2.

```
> Review_data$Review = gsub('[[:punct:]]', '', Review_data$Review)
> Review_data$Review = tolower(Review_data$Review)
> View(Review_data)
> write.csv(Review_data, "Cleaned_Review.csv")
>
```

Figure 2: R functions for data cleaning

Online_Reviews	Large character (689.1 Kb)
Online_Reviews_basic_lower	Large character (674 Kb)
Online_Reviews_basic_punct	Large character (674 Kb)

Figure 3: Changes in data size

After using these functions, the noise in the data was reduced by almost 15kb, as shown in figure 3. This technique has a few shortcomings like, it doesn't remove all the special symbols, and lacks a function to remove the stop words and white spaces. To address these shortcomings another method was used which involves natural language processing library packages like TM Library or Text mining Library package. It is a text mining framework to provide various functions that help in text pre-processing. It involves different ways for importing data, handling the corpus and preprocessing methods. A corpus that is a collection of text documents is created for managing the documents in tm package.

Unlike the traditional method, the data cannot be imported in the table format so it needs to be converted into an array by using a vector source function to create a Corpus and then a pre-defined library called Tm-map was used to clean the data in phases, figure 4.

4. RESULTS

The table below shows the analysis of the traditional methods and NLP Library packages for data pre-processing. It shows that traditional methods only facilitates the conversion of text in lower case and removal of punctuation marks which reduces the data size to 674 kb from 689 kb by removing 15 kb of noise.

```
write.csv(Review_data, "Cleaned_Review.csv")
# ..................................
Documet_clean <- Corpus(VectorSource(Online_Reviews))
#inspect(Documet_clean)
toSpace <- content_transformer(function (x , pattern ) gsub(pattern, " ", x))
Documet_clean <- tm_map(Documet_clean, toSpace, "/")
Documet_clean <- tm_map(Documet_clean, toSpace, "@")
Documet_clean <- tm_map(Documet_clean, toSpace, "\\|")

# Convert the text to lower case
Documet_clean <- tm_map(Documet_clean, content_transformer(tolower))

# Remove numbers
#Documet_clean <- tm_map(Documet_clean, removeNumbers)

# Remove english common stopwords
Documet_clean <- tm_map(Documet_clean, removeWords, stopwords("english"))

# Remove your own stop word
# Specific words to remove
Documet_clean <- tm_map(Documet_clean, removeWords, c("blabla1", "blabla2"))

# Remove punctuations
Documet_clean <- tm_map(Documet_clean, removePunctuation)

# Eliminate extra white spaces
Documet_clean <- tm_map(Documet_clean, stripWhitespace)
Documet_clean <- TermDocumentMatrix(Documet_clean)
```

Figure 4: Tm Library for data cleaning

Table 1: Merits of using NLP library packages for text as compared to Traditional methods.

S.No	Traditional Methods	NLP Library Packages
1	Removes Punctuation marks	Removes Punctuation marks
2	Converts the text in lower case	Converts the text in lower case
3	Doesnot eliminate stop words	Eliminates the stop words
4	Doesnot help in removing white spaces	Eradicates the white spaces
5	Doesnot remove special characters	Eliminates the special characters
6	Helps to eliminate around 15kb noise from data	Eliminated around 200kb noise from data

NLP library packages facilitate many more functionalities like removing special characters, converting the text in lower case, removal of punctuation marks, stop word removal and white space removal and hence reduced the data size to 445 kb, thus removing 200kb noise from the data, as shown in figure 5. The results show that when the latter process is used, the performance of the tool will be better as classification process will be more precise.

5. CONCLUSION

This paper focuses on the most important step for sentiment analysis, which is the pre-processing of customer reviews in R and illustrates two methods used for it. The results show that use of NLP Library packages enhances the text pre-processing when used for sentiment analysis of customer reviews as provided by the traditional methods. This finding will be useful for organizations that work on opinion mining to extract the sentiments portrayed in the reviews written in the form of text. It is clear from the results that using library packages is a preferred approach for performing the pre-processing of text before it can be analyzed.

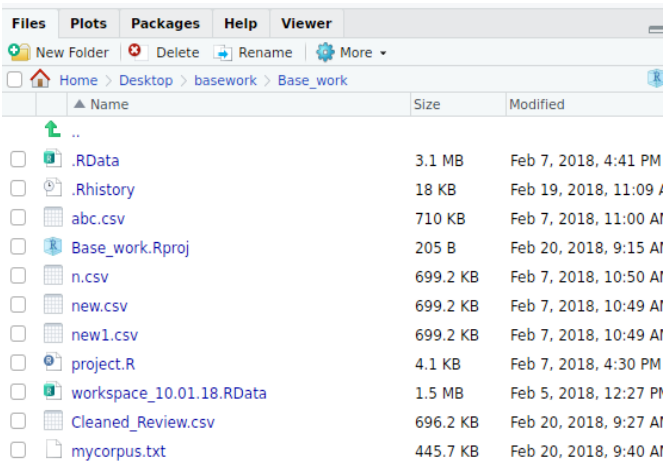

Figure 5: Change in size of data after using TmMap

6. REFERENCES

[1] Tang, H., Tan, S., Chang, X. 2009. A Survey on sentiment detection of reviews. Expert Systems with Applications 36(7)(2009) 10760-10773

[2] Thelwall, M., Buckley, K., Paltoglou, G. 2011. Sentiment in twitter events.Journal of the American Society for Information Science & Technology 62(2) (2011) 406-418.

[3] Riloff, E., Patwardhan, S., and Wiebe, J. 2006. Feature subsumption for opinion analysis. In Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, pages 440–448. Association for Computational Linguistics, 2006.

[4] Chen, Z., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M., and Ghosh, R.2013. Exploiting domain knowledge in aspect extraction. In EMNLP, pages 1655–1667. 2013.

[5] Li Lee Mong.1999. Cleansing Data for Mining and Data warehousing. School of computing National University of Singapore, 1999 .

[6] Rahm E. & Hai Do Hong. 2000. Data Cleaning: Problems and current approaches. IEEE Bulletin of the Technical Committee on Data Engineering, 2000.

[7] Ibrahim Housien Hamed, Zuping Zhang & Qays Abdulhadi Zainab.2013. A comparison study Of Data Scrubbing algorithm and framework in Data Warehousing. International Journal of Computer Applications (0975 – 8887) April 2013.

[8] Choudhary Nidhi.2014. A Study over Problems and Approaches of Data Cleansing/Cleaning,. Volume 4, Issue 2, February 2014 .

[9] Hellerstein Joseph, M. 2008. Quantitative cleaning of large databases February 27,2008.

[10] Y.Patil Rajashree, Dr. Kulkarni R.V.2012. A Review of Data Cleaning Algorithms for Data Warehouse

Systems. IJCSIT , Vol. 3 (5) , 2012.

[11] Müller Heiko & Christoph Freytag Johann. Problems, Methods, and Challenges in Comprehensive Data Cleansing. Humboldt-Universität zu Berlin zu Berlin,10099 Berlin, Germany.

[12] Li Lee Mong, Wang Ling Tok & Lup Low Wai.2000. IntelliClean: A knowledge-based intelligent data cleaner, Proceedings of the ACM SIGKDD, Boston, USA, 2000.

[13] Sarpong Kofi Adu-Manu, Davis Joseph George, Panford Joseph Kobina.2013. A Conceptual Framework for Data Cleansing – A Novel Approach to Support the Cleansing Process. International Journal of Computer Applications, Volume 77– No.12, September 2013.

[14] Peng Taoxin. A framework for data cleanings in data warehouses. School of computing, napier University,10 Colinton Road, Edinburgh, EH10 5DT, UK.

[15] I.Fienerer, K.Hornik, D.Meyer.2008. Text mining infrastructure in R. Journal of Statistical Software 25 (5) (2008) 1 54.