Text Mining and Sentiment Analysis

Shraddha Shekhar

Student, Dept. Computer Engineering, Sinhgad College of Engineering, Maharashtra, India

---***---

Abstract - *As a huge amount of data is generated every-day and this data is in unstructured format, Text Mining is one of the fastest growing technology for extraction of information from this unstructured data. With the help of NLP (natural language processing) Text Mining is simplified. This report consists of, study of text mining, its implementation using NLP with the help to nltk, a text toolkit provided by python for NLP. It also specifies study of Sentiment analysis. In a course of time, a large number of audits are created on the web about an item, individual or a place. Sentiment Analysis is a research area which comprehends and extricates the assessment from the given review and the analysis process in corporates natural language processing (NLP), computational linguistics, text analytics and classifying the polarity of the opinion. In the field of sentiment analysis, there are numerous algorithms exist to tackle.*

Key Words: (NLTK, Text Mining, Sentiment Analysis, Python)

1. INTRODUCTION

Retrieving information is just a matter of seconds now-a-days, but ever thought How does this happen? How is this data retrieved? Around 43 zettabytes of data is generated every-day, and about 89% of this data is trivial in the analysis process. Extracting useful information out of this raw data is nothing but Text Mining. Ever got an advertise pop-up in the middle of surfing on the internet related to what you previously searched for? Or how does You-tube select "recommended" videos for you? This is nothing but an application of Text Mining, more specifically sentiment analysis. Text mining simply means analyzing the huge data stored in the database, and extracting the relevant and required information from it for further processing. And sentiment analysis means evaluating the text for the tone of the text, whether it is positive, negative or neutral. Mining of data in any forms (text, image, audio, video) is solely carried out for extracting necessary information from the raw data. In this report you will find the study of Text mining and Sentiment Analysis, what it exactly is and how it is actually implemented, from scratch.

Section 1.1 discusses about Motivation for the topic, Section 1.2 about the Timeline Evolution a and Section 1.3 states the Organization flow of the report.

1.1 Motivation

A huge amount of data is generated every-day, and most of the data is held in unstructured format, most of which is trivial for analysis process. Examining this large data and processing it for extracting the necessary information, is a very tedious and exhaustive task. This is where Text Mining comes in the picture.

For the purpose of extracting valuable and non-trivial information from this unstructured data, Text mining has become a need in today's time. Providing with different facilities for almost all requirements in data processing, text mining makes the task a lot easier and saves the time. With the help of NLP (natural language processing), text mining has got a new level of simplicity.

Curiosity of knowing how this conversion happens and how this huge data is handled, motivated me to choose this topic.

1.2 Organization of the report

Chapter 1 is the introductory section.

Chapter 2 discusses the Literature review and its fundamentals.

Chapter 3 contains following topics:

1. Introduction to Text Mining
2. Introduction to text mining using NLP
3. Terminologies of NLP
4. Applications of Text Mining
5. Introduction to Sentiment Analysis
6. Process of SA
7. Application of SA

Chapter 4 contains the results and outcomes of the seminar.

Chapter 5 is the concluding section and also some future scope of the topic.

2. METHODOLOGY

This chapter includes the explanation for all the topics and its subtopics that the report comprises of. Section 3.1 discusses about Text Mining and all its sub topics. Section 3.2 explains NLP and its terminologies. Section 3.3 explains how machine learning can be applied and Section 3.4 gives the application of Text Mining. The next section 3.5 introduces you to Sentiment analysis and its process and Section 3.6 explains Sentiment Classification. Lastly Section 3.7 jots down the applications of SA and Section 3.8 provides a simple example of Twitter Sentiment Analysis.

2.1 Text Mining

Text Mining, aka intelligent text analysis, is an automatic process that uses natural language processing to extract valuable insights from unstructured text. By transforming data into information that machines can understand, text mining automates the process of classifying texts by sentiment, topic, and intent.

A huge amount of data is generated every day and most of the data is trivial. Extracting useful information out of this raw data has become the need in today's world.

Different techniques of Text mining:

- Information Extraction
- Information Retrieval
- Summarization
- Clustering
- Categorization

2.1.1 Information Extraction

The primary stage for computers is to recognize amorphous typescript by recognizing important phrases and dealings within text. It aims at extracting meaningful information from huge chunk of text. The information mined is preserved in the form of a record for further access or recovery. The practice may produce the different results subject to the purpose of the process and elements of textual data. The scope of information is defined by the elements of the textual data. These elements are tokens terms and separators can be termed as tokens. Characters like blank space or a punctuation mark are considered as separators. A token with specific semantic purpose can be a defined as a term. Various methods of extracting information can be token extraction, entity extraction, term parsing and complex fact extraction.

2.1.2 Information Retrieval

The field of Information retrieval has been under construction with database systems for more duration. The aim of Information Retrieval is to gain the document with precise information retrieved by the user. Thus recovery of document is followed by stage named as text mining which mainly focuses on request posted by the operator or it is followed by information extraction stage which uses information extraction techniques. Information retrieval, also known as IR, mainly focuses on searching and retrieving a document. Every search query has a response, in this response, out of collection of documents are retrieved. Here, the unsaturated textual data is present in large amount in a single document.

2.1.3 Summarization

Summarization is collecting and producing concise representation of documents with original text, this process is called as text summarization. In summarization, first the raw text is taken and pre-processing and processing operations are performed on it. In pre-processing, three methods are applied i.e., tokenization stemming and word removal methods are applied. At handling stage of text summarization, generation of lexicon lists take place. The performance of automatic text summarization was influenced by rate of appearance of words or phrases in the last few years. Later to increase correctness of results some more methods were brought into practice with the standards procedure of text mining. Multiple documents can apply text summarization techniques at the same time. The subject of the documents depends on the quality and type of classifiers. Precise text is generated from number of documents in Summarization. It is not often possible to encapsulate huge textual file. Also, in centres used in for examining all the documents cannot be read. They basically summarize documents and make up the summary of document from important points.

2.1.4 Clustering

The process of sectioning a group of objects or data into collection of relevant and understandable subclasses is termed as clustering. Clustering is mainly used to make a set of similar documents and files. The advantage of clustering is that the document or text files will be in multiple sub topics, which makes it safer for important documents from getting erased from search. Clustering technique separates records in a dataset into groups in such a way that themes in a cluster are same while themes between the clusters are different. Acquiring the group that has some value with regard to the difficulty being addressed is the main aim of cluster analysis. The result is not achieved always. Clustering is mainly classified into two types:

i. Hierarchical
ii. Non hierarchical

2.1.5 Categorization

In categorization, the important themes of a document are recognized. This is done by assigning the documents into a set of topics which are predefined. The document which is categorized it can be treated as 'bag of words'. As information extraction does attempt to process the actual information whereas categorization doesn't attempt to process the actual information. In this the words of the document are counted by categorization process then using the counts they recognize the important subjects of the documents.

2.2 NLP (Natural Language Processing)

NPL is a field of AI in which computer analyzes, understands and derive meaning of human language in a smart and useful way. The history of natural language processing (NLP) generally started in the 1950s, although work can be found from earlier periods. In 1950, Alan Turing published an article titled "Computing Machinery and Intelligence" which proposed what is now called the Turing test as a criterion of intelligence. Up to the 1980s, most natural language processing systems were based on complex sets of hand-written rules. Starting in the late 1980s, however, there was a revolution in natural language processing with the introduction of machine learning algorithms for language processing. It is the finest of all for Text Mining. It is widely used with the Natural Language Tool Kit provided by python. [8]

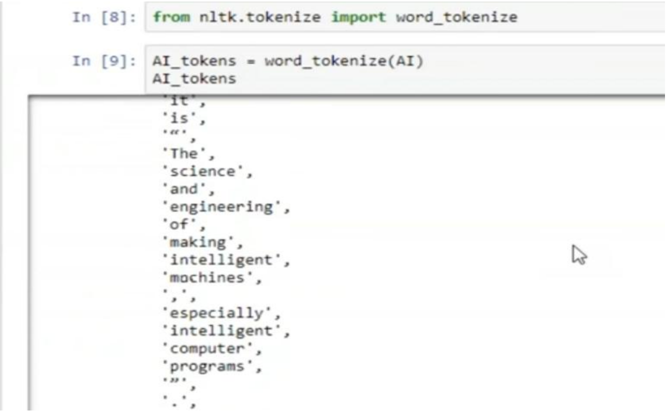
The algorithm consists of following steps:

1. Tokenization
2. Stemming
3. Lemmatization
4. Data Cleaning
5. POS Tags
6. Named Entities
7. Chunking

2.2.1 Tokenization

It is the first step in NLP and consist of 3 steps:

- Break a complex sentence into words.
- Understand the importance of each word with respect to the sentence.
- Produce a structural description on an input sentence.



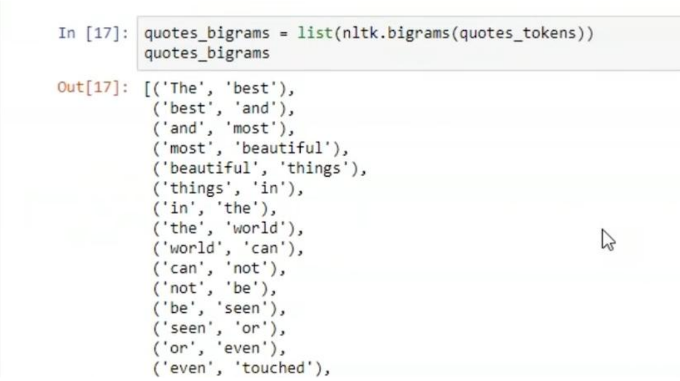
```
In [8]:  from nltk.tokenize import word_tokenize

In [9]:  AI_tokens = word_tokenize(AI)
         AI_tokens
          'it',
          'is',
          '“',
          'The',
          'science',
          'and',
          'engineering',
          'of',
          'making',
          'intelligent',
          'machines',
          ',',
          'especially',
          'intelligent',
          'computer',
          'programs',
          '”',
          '.',
```

Fig. (2.2.1) Tokenization using nltk

Using word_tokenize() function form nltk.tokenize the sentence is converted to a list of tokens/words which are used for further processing. Using FreqDist() from nltk.probability we can find the frequency of every token.

2.2.1.1 Bigrams, Trigrams and Ngrams:

Bigram gives a list of tuples containing 2 consecutive tokens from the tokenized list. Similarly Trigram produces tuples of 3 consecutive tokens and Ngram for N tokens.



```
In [17]:  quotes_bigrams = list(nltk.bigrams(quotes_tokens))
          quotes_bigrams

Out[17]:  [('The', 'best'),
           ('best', 'and'),
           ('and', 'most'),
           ('most', 'beautiful'),
           ('beautiful', 'things'),
           ('things', 'in'),
           ('in', 'the'),
           ('the', 'world'),
           ('world', 'can'),
           ('can', 'not'),
           ('not', 'be'),
           ('be', 'seen'),
           ('seen', 'or'),
           ('or', 'even'),
           ('even', 'touched'),
```

Fig (2.2.2) Bigrams using nltk

2.2.2 Stemming

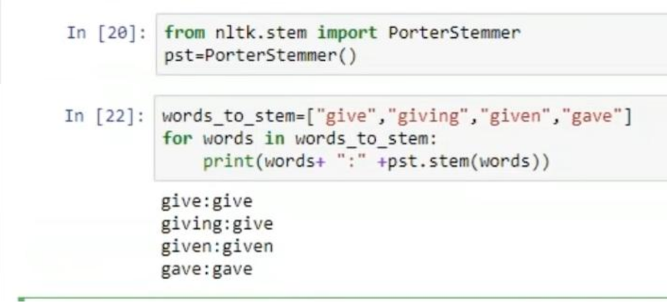
Stemming is Normalizing words into its base form or root form. For eg.

Detected Detection Detecting Detections has the root form Detect.

Stemming algorithm works by cutting off the end or the beginning of the word by taking in account the common set of suffixes and prefixes and hence the actual root form of the word may not be obtain every time.

There are 2 main types of stemming, PorterStemmer and LancasterStemmer LancasterStemmer is more robust than the PorterStemmer.

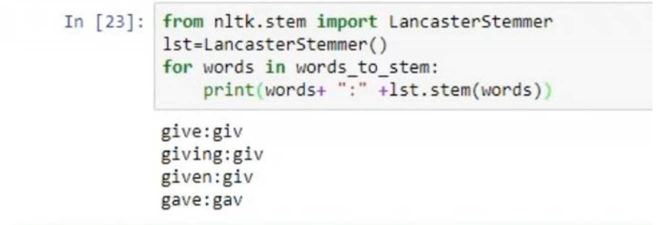
- PorterStemmer



```
In [20]:  from nltk.stem import PorterStemmer
          pst=PorterStemmer()

In [22]:  words_to_stem=["give","giving","given","gave"]
          for words in words_to_stem:
              print(words+ ":" +pst.stem(words))

          give:give
          giving:give
          given:given
          gave:gave
```

Fig (2.2.3) PorterStemmer using nltk

- LancasterStemmer



```
In [23]:  from nltk.stem import LancasterStemmer
          lst=LancasterStemmer()
          for words in words_to_stem:
              print(words+ ":" +lst.stem(words))

          give:giv
          giving:giv
          given:giv
          gave:gav
```

Fig (2.2.4) LancasterStemmer using nltk

As in the above example it is clear that LancasterStemmer is more aggressive as it gives the root for more form of words. But the root obtained is not an actual meaningful word every-time.

2.2.3 Lemmatization

This step Groups together different inflected forms of words, called Lemma.

It is somewhat similar to stemming as it maps several words having same meaning of root Output is a proper word. A detail dictionary is required as it maps similar words which is provided by WordNet from nltk.stem. Lemmatization also streamlines the analysis process by removing redundant data by mapping them to the common root.

eg. Gone, Going and went is mapped to the same root Go.

2.2.4 Data Cleaning

Removing Stopwords and Punctuations and all other unnecessary tokens form to make the processing easy. Stopwords are words that are trivial for processing and needs to be removed for more precise analysis.

- Algorithm:

```
from nltk.corpus import stopwords
import re
punctuations=re.compile(r'[-,?:;()|0-9]')
final=[]
for words in AI_tokens:
        word=punctuations.sub("",words)
        if len(word)>0:
                final.append(word)
```

This removes the punctuations from list of tokens.

2.2.5 POS Tags (Parts Of Speech)

These are the Tags given to every token according to the grammar rules. It describes the token and classifies it by assigning respective tags.

Eg. CC-Coordinating Conjunction, DT-Determiner, VB-Verb, JJ-Adjective, etc.

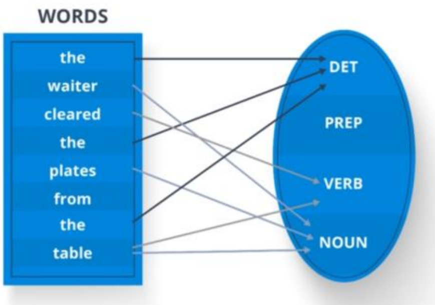
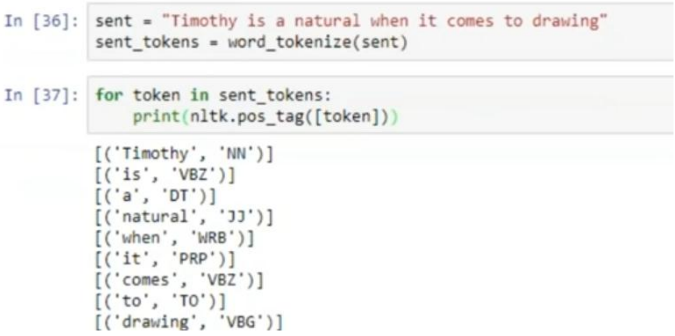


Fig (2.2.5) POS Tags assignment



```
In [36]: sent = "Timothy is a natural when it comes to drawing"
         sent_tokens = word_tokenize(sent)

In [37]: for token in sent_tokens:
             print(nltk.pos_tag([token]))

         [('Timothy', 'NN')]
         [('is', 'VBZ')]
         [('a', 'DT')]
         [('natural', 'JJ')]
         [('when', 'WRB')]
         [('it', 'PRP')]
         [('comes', 'VBZ')]
         [('to', 'TO')]
         [('drawing', 'VBG')]
```

Fig (2.2.6) POS Tags assigned using nltk

2.2.6 Named Entity Recognition

Any movie name, organization name, Monetary value, Quantities, location, person name is called named entity recognition.

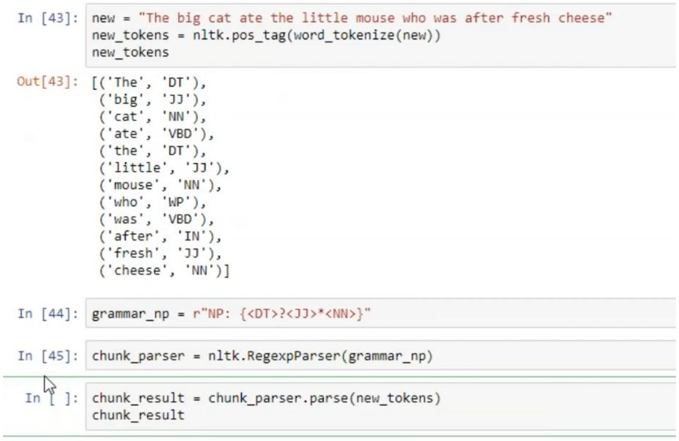
It recognizes the named entity and labels it with appropriate tag.

Groups the words sometimes, if required.

2.2.7 Chunking

Chunking means picking up individual pieces of information and grouping them into bigger pieces, known as chunks. It groups the Noun Phrases according to the specified grammar.

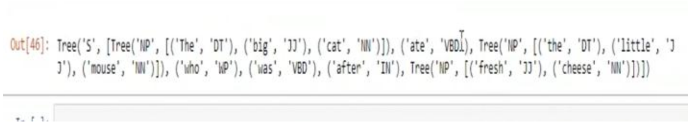
In the example we first specify a Regular Expression. Then a parser is created according to the Regular Expression. The text/doc is firstly tokenized, and then these tokens is are passed to the parser that we generated to form the tree.



```
In [43]: new = "The big cat ate the little mouse who was after fresh cheese"
         new_tokens = nltk.pos_tag(word_tokenize(new))
         new_tokens

Out[43]: [('The', 'DT'),
          ('big', 'JJ'),
          ('cat', 'NN'),
          ('ate', 'VBD'),
          ('the', 'DT'),
          ('little', 'JJ'),
          ('mouse', 'NN'),
          ('who', 'WP'),
          ('was', 'VBD'),
          ('after', 'IN'),
          ('fresh', 'JJ'),
          ('cheese', 'NN')]

In [44]: grammar_np = r"NP: {<DT>?<JJ>*<NN>}"

In [45]: chunk_parser = nltk.RegexpParser(grammar_np)

In [ ]: chunk_result = chunk_parser.parse(new_tokens)
        chunk_result
```

Fig (2.2.7) Chunking using nltk

The output for chunking is a tree of Noun Phrases (as shown below) which clubs together the noun phrases which will be formed according to the regular expression that we specify in the code.



```
Out[46]: Tree('S', [Tree('NP', [('The', 'DT'), ('big', 'JJ'), ('cat', 'NN')]), ('ate', 'VBD'), Tree('NP', [('the', 'DT'), ('little', 'J
J'), ('mouse', 'NN')]), ('who', 'WP'), ('was', 'VBD'), ('after', 'IN'), Tree('NP', [('fresh', 'JJ'), ('cheese', 'NN')])])
```

Fig (2.2.8) Output of Chunking (NP tree)

2.3 Applying Machine Learning

This example represents how Naïve Bayes algorithm can be used and its pros and cons. Naïve Bayes classifier is a simple probabilistic classifier based on applying Bayes' Theorem with strong independence assumptions. A more descriptive term for the underlying probability model would be independent feature model. These independence assumptions of features make the features order is irrelevant and consequently that the present of one feature does not affect other features in classification tasks [10]. These assumptions make the computation of Bayesian classification approach more efficient, but this assumption severely limits its applicability. Depending on the precise nature of the probability model, the naïve Bayes classifiers can be trained very efficiently by requiring a relatively small amount of training data to estimate the parameters necessary for classification. Because independent variables are assumed, only the variances of the variables for each class need to be determined and not the entire covariance matrix. Due to its apparently over-simplified assumptions, the naïve Bayes classifiers often work much better in many complex real-world situations than one might expect. The naïve Bayes classifiers has been reported to perform surprisingly well for many real world classification applications under some specific conditions [11] [12] [13] [14] [15]. An advantage of the naïve Bayes classifier is that it requires a small amount of training data to estimate the parameters necessary for classification. Bayesian classification approach arrives at the correct classification as long as the correct category is more probable than the others. Category's probabilities do not have to be estimated very well. In other words, the overall classifier is robust enough to ignore serious deficiencies in its underlying naïve probability model. The main disadvantage of the naïve Bayes classification approach is its relatively low classification performance compare to other discriminative algorithms, such as the SVM with its outperformed classification effectiveness. Therefore, many active researches have been carried out to clarify the reasons that the naïve Bayes classifier fails in classification tasks and enhance the traditional approaches by implementing some effective and efficient techniques [11] [13] [14] [15] [16].

$$P(c_i \mid D) = \frac{P(c_i)P(D \mid c_i)}{P(D)}$$

$$P(D \mid c_i) = \prod_{j=1}^{n} P(d_j \mid c_i)$$

Where P(Ci)= $P(C = c_i) = \frac{Ni}{N}$

and P(dj|ci) = $P(d_j \mid c_i) = \frac{1 + N_{ji}}{M + \sum_{k=1}^{M} N_{ki}}$

Naïve Bayes has been one of the popular machine learning methods for many years. Its simplicity makes the framework attractive in various tasks and reasonable performances are obtained in the tasks although this learning is based on an unrealistic independence assumption. For this reason, there also have been many interesting works of investigating naive Bayes. Recently the [17] shows very good results by selecting Naïve Bayes with SVM for text classification also the authors in [18] prove that Naive Bayes with SOM give very good results in clustering the documents.

2.4 Applications of Text Mining

2.4.1 Advertising

Advertising is a field which is implementing text mining on a large scale and is rapidly growing. Text. The main goal of advertising is to find the most appropriate audience for their product to increase their sales. Text mining techniques are implied on the searches of the user and accordingly the most relevant ads pop up on the screen of the user. Text Mining is immensely applied in advertising as is very helpful in increasing the business productivity.

2.4.2 Digital Libraries

There are many techniques and tools of text mining which are used to set up patterns and trends from journals and its proceedings from massive amount of sources. For doing research, we can get great source of information for the research and the one making an effort to the significance are digital libraries. It offers interestingly new method of organizing information in such a way that it is available in trillions of documents online. It provides an innovative way to organize information and access it to millions of documents which are available online. Greenstone international digital libraries provides a springy method for extract documents of multiple formats such as MS Word, PDF, postscripts, HTML, script languages and email messages and it also supports multiple languages and multilingual interface. Various operations such as document selection, enrichment, information extraction and tackling articles in the middle of the documents are performed. The most frequently used tool for mining in digital libraries is Gate and Net Owl.

2.4.3 Business Intelligence

Organizations and enterprises use text mining to analyse their customer and competition to take superior decision. It has a greater significance as it provides vision about the business and provides customer satisfaction and get more competition advantages. Text mining tools like such as IBM, text analysis, rapid mine, Gate supports to make conclusion about the institute that warns about the positive and negative execution.

2.4.4 Social Media

Packages in Text mining software are able to access any social media applications and observe or learn the script from social media, blogs, internet, news or email etc. Text mining tools are very helpful in recognizing or analysing total number of followers, post and likes on the social media. Due to this type of scrutiny it becomes very easy to find out people's response on various news, post and how it expands socially. It may even show the behaviour of different people who fit in to precise age crowd or people having similar opinions or variation about the similar post.

2.5 Sentiment Analysis

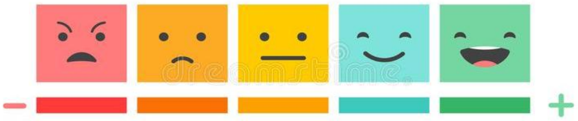


Fig (2.5.1) Sentiments
(Courtesy: Google)

Sentiment analysis is the task of identifying positive and negative opinions, emotions, and evaluations. Sentiment analysis is the process of extracting emotions or opinions from a piece of text for a given topic. It allow us to understand the attitudes, opinions and emotions in the text. In it user‘s likes and dislikes are captured from web content. It involves predicting or analyzing the hidden information present in the text. This hidden information is very useful to get insights of user's likes and dislikes. The aim of sentiment analysis is to determine the attitudes of a writer or a speaker for a given topic. Sentiment analysis can also be applied to audio, images and videos. [19] Today internet has become the major part of our life. Most of the people use online blogging sites or social networking sites to express their opinions on certain things. They also use these sites to know what other people's opinions are. Thus, mining of this data and sentiment extraction has become an important field of research.

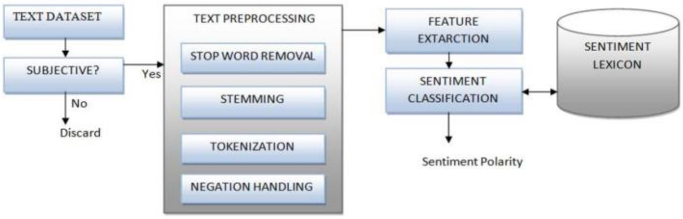


Fig (2.5.2) Sentiment Analysis Process
(Courtesy: Google)

A) Feature selection: To perform sentiment classification, first task is to extract the features from text which are
1) N grams- n grams refer to consecutive n terms in text. One can take only one word at a time(unigram) or two words(bigram) up to n accordingly. Some sentiments can't be captured with unigram feature. For example, this drink will knock your socks off. It is a positive comment if socks off is taken together and negative in case of only unigram model (off).

2) POS tagging- It is a way toward denoting a word in a content (corpus) as comparing to parts of speech in light of both its definition and its association with contiguous words. Nouns, pronouns, adjectives, adverbs etc are examples parts of speech. Adjectives and adverbs hold most of the sentiments in text.

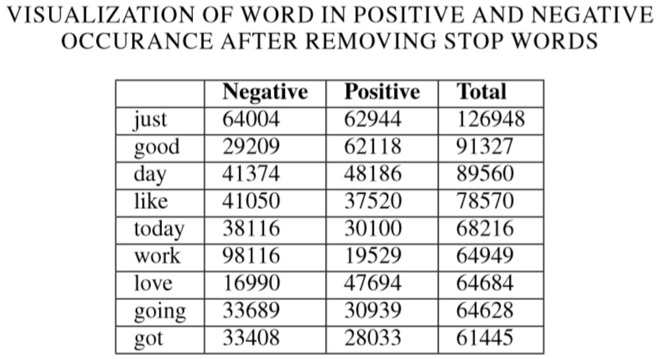
3) Stemming-It is the process of removing prefixes and suffixes. For example 'playing', 'played' can be stemmed to 'play'. It helps in classification but sometimes leads to decrease in classification accuracy.

4) Stop words- Pronouns (he/she, it), articles (a, the), prepositions (in, near, beside) are stop words. They provide no or little information about sentiments. There is a list of stop words available on the internet. It can be used to remove them in the pre-processing step.

5) Conjunction handling- In general, each sentence expresses only one meaning at a time. But certain conjunction words like but, while, although, however changes the whole meaning of the sentence. For example although movie was good but it was not up to my expectations. By using these rules throughput can be increased by 5%.

6) Negation handling- Negation words like 'not' inverts the meaning of whole sentence. For example the movie was not good has 'good' in it which is positive but 'not' inverts the polarity to negative. [20]

The polarity of the text is calculated according to the positive, negative values assigned to the word/token. The sum is calculated and final polarity is displayed. An example specifying the format of calculation is shown below.



VISUALIZATION OF WORD IN POSITIVE AND NEGATIVE OCCURANCE AFTER REMOVING STOP WORDS

	Negative	Positive	Total
just	64004	62944	126948
good	29209	62118	91327
day	41374	48186	89560
like	41050	37520	78570
today	38116	30100	68216
work	98116	19529	64949
love	16990	47694	64684
going	33689	30939	64628
got	33408	28033	61445

Table (2.5.3) Polarity Table [3]

2.6 Sentiment Classification

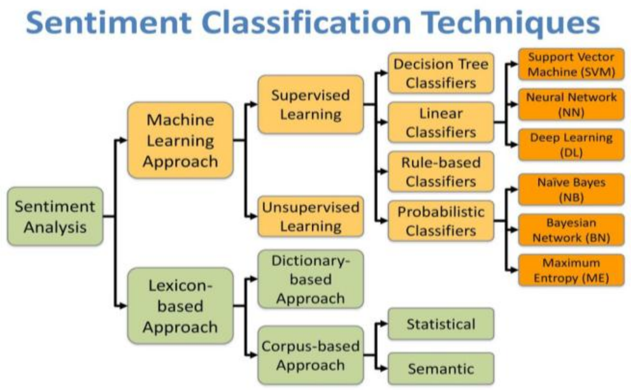


Fig (2.6.1) Sentiment Classification Techniques (Courtesy: Google)

Two approaches are mainly used:

2.6.1 Subjective lexicon

Subjective lexicons are collection of words where each word has a score indicating the positive, negative, neutral and objective nature of text. In this approach, for a given piece of text, aggregation of scores of subjective words is performed i.e. positive, negative, neutral and objective word scores are summed up separately. In the end there are four scores. Highest score gives the overall polarity of the text. Sentiment of the given text in represented by polarity and subjectivity. The polarity measures how positive or negative in text is on a scale of (-1,0,1) and subjectivity represents how much of an opinion it is vs how much of a fact.

2.6.1.1 Dictionary based approach- In this approach a set of opinion words are manually collected and a seed list is prepared. Then we search for dictionaries and thesaurus to find synonyms and antonyms of text. The newly found synonyms are added to the seed list. This process continues until no new words are found. Disadvantage: difficulty in finding context or domain-oriented opinion words

2.6.1.2 Corpus based approach- Corpus is collection of writings, often on a specific topic. In this approach, seed list is prepared and is expanded with the help of corpus text. Thus is solves the problem of limited domain-oriented text. It can be done in two ways

a) Statistical approach: This approach is used to find cooccurrence words in the corpus. Idea is that if the word appears mostly in positive text, then its polarity is positive. If it mostly occurs in negative text, then its polarity is negative.

b) Semantic approach: This approach calculates sentiment values by using the principal of similarity between words. Wordnet can be used for this purpose. Synonyms and antonyms of given word can be found using this and sentiment value can be calculated.

2.6.2 Machine learning

This is an automatic classification technique. Classification is performed using text features. Features are extracted from text. It is of two types- supervised and unsupervised.

a) Supervised- System is trained using labeled training examples. Each class represents different features and has a label associated with it. When a word arrives, its features are compared and labeled with a class with maximum matching.
1. Probabilistic classifier: This classifier is able to foresee a probability function over a set of classes for a given input data. It does not give only the most likely classes but a probability function over all classes. For example an ordinary classifier function assign a label y to input x as y=f(x)

In case of probabilistic classifier this function is replaced with conditional distributors Pr(Y/X) i.e. for given x X , probability is assigned to all y Y as y= arg max Pr(Y=y/X)

Naïve bayes: This classifier uses bayes theorem to predict the probability that a given set of features is a part of particular label. It uses bag of words (BOW) model for feature extraction. This model assumes that all the features are independent.
P(label/features)=P(label)* P(features/label)/P(features)

Where P(label)= prior probability of label P(features/label)=prior probability that feature set is classified as label P(features)= prior probability that feature set will occur.
Many other methods are discussed in [20]

2.7 Applications of Sentiment Analysis

1. On of the famous application of sentiment analysis is Twitter Sentiment Analysis, analyzing the tweets containing a particular word and determining its sentiment is carried out on a large scale for many surveys. By obtaining the polarity it is easy to determine the sentiment.
2. Increasing the business by understanding customer need, likes and dislikes is conducted using sentiment analysis hugely today.

3. Customer Care services now a day use sentiment analysis for evaluating the customer reviews without reading them all and providing the necessary changes as per the customer's review. Getting feedbacks on a product and finding out the views on the product from the customer point of view.

3. RESULT AND DISCUSSION

A good understanding regarding the topics explained, TM, SA, NLP is incurred from the report.

All the information required to begin with Text Mining, all the terminologies of NLP are explained in detail. Along with it the actual implementation with python using nltk and various facilities provided for text processing are discussed with proper coding. Code snippets for the functions are provided so as to understand the working and output for each.

Introduction to Text Mining and Sentiment Analysis and their need is discussed. How these technologies are evolving and the applications they are used in are stated. Text mining allows processing of unstructured data which contributes in a lot of applications for information and knowledge extraction which is massively immerging in BI to expand the business empire.

Furthermore, opinion mining and sentiment analysis also reviewed. During the review it has been analysed that huge of NLP techniques are available for opinion mining and sentiment analysis. The fundamental point of opinion mining and sentiment analysis is extracting presence of sentiments from the given writings. To process the given undertaking feeling mining could be separated into three dimensions: document level, sentence level, and fine-grained level.

Use of SA is in every field and yet growing. As we discussed how it works and where it is used the results are prominent in the fields of analyzing the sentiments in the text and providing the polarity of the text and how much of positive or negative or neutral it is.

With a small and widely used example we discussed the application in Twitter sentiment analysis. Many more applications are implementing Text Mining and Sentiment Analysis making them rapidly growing technologies.

Text Mining has come challenges:

- Identification of spam, low value content, and users with multiple accounts. Need to segment and score users.
- Potential privacy or liability issues, e.g. if data gathered is used to target people individually, through marketing campaigns, fraud investigations or to penalize users (e.g. refusing a job to a candidate based on data mining of user posts on social networks).[22]
- Sarcasm detection is difficult as it includes synonymous meanings for some words and correct interpretation of the word is not accurate all the time.

4. CONCLUSIONS

Text mining techniques help in deriving different traits from amorphous textual data. Several methods and techniques lead to well-organized and accurate text mining. This report is based on how mining should be performed on textual data. The process of text mining, its applications, Information retrieval, Summarization and various such methods have been discussed. A very convincing approach is discovered due to observations, because of which methods are examined and upgrading of method is suggested.

The implementation using python's nltk is discussed and explained. How the tools and it's facilities can be utilized to perform text mining using NLP as discussed.

A general approach for Sentiment Analysis is discussed and different approaches are explained.

An overall description of NLP, its terminologies, its application for text mining, Sentiment Analysis, its classification and its applications are conversed and explained. It is a descriptive report for getting introduced to NLP, TM, SA.

Two basic approaches [Limited accuracy, depth]

– Statistical Signature of Bag of Words
– Dictionary of positive & negative words. [21]

5. FUTURE WORK

The scope of Text Mining is very promising in the future as the amount of Text Data is increasing exponentially day by day. Social media platforms are generating a lot of text data which can be mined to get real insights about different domains.

Emotion taxonomies - Joy, Sadness, Fear, Anger, Surprise, Disgust – New Complex – pride, shame, embarrassment, love, awe – New situational/transient – confusion, concentration, skepticism

Analysis of Conversations- Higher level context – Techniques: self-revelation, humor, sharing of secrets, establishment of informal agreements, private language – Detect relationships among speakers and changes over time – Strength of social ties, informal hierarchies

Importance of Context – around positive and negative words – Rhetorical reversals – "I was expecting to love it" – Issues of sarcasm, ("Really Great Product").

New types of applications – New ways to make sense of data, enrich data.

Harvard – Analyzing Text as Data
– Detecting deception, Frame Analysis
Narrative Science – take data (baseball statistics, financial data) and turn into a story

REFERENCES

[1] https://thesai.org/Downloads/Volume7No11/Paper_53-Text_Mining_Techniques_Applications_and_Issues.pdf

[2] "Marti Hearst: What is Text Mining?"

[3]https://www.researchgate.net/publication/334167408_Machine_Learning_Based_Approach_To_Sentiment_Analysis

[4]https://www.semanticscholar.org/paper/Natural-Language-Processing-and-Text-Mining-to-for-Valdez-Almada-Rodr%C3%ADguez-Elias/70e46639f3f9a570797fd6adf337f722bf0330bf

[5]https://www.researchgate.net/publication/220355220_Recognizing_Contextual_Polarity_An_Exploration_of_Features_for_Phrase-Level_Sentiment_Analysis

[6]https://www.researchgate.net/publication/330796490_Review_on_Natural_Language_Processing_NLP_and_Its_Toolkits_for_Opinion_Mining_and_Sentiment_Analysis

[7]Hotho, A., Nürnberger, A. and Paaß, G. (2005). "A brief survey of text mining". In Ldv Forum, Vol. 20(1), p. 19-62

[8]https://en.wikipedia.org/wiki/Natural_language_processing

[9]Heide Brücher, Gerhard Knolmayer, Marc-André Mittermayer; "Document Classification Methods for Organizing Explicit Knowledge", Research Group Information Engineering, Institute of Information Systems, University of Bern, Engehaldenstrasse 8, CH - 3012 Bern, Switzerland. 2002.

[10]Andrew McCallum, Kamal Nigam; "A Comparison of Event Models for Naïve Bayes Text Classification", Journal of Machine Learning Research 3, pp. 1265-1287. 2003.

[11]Irina Rish; "An Empirical Study of the Naïve Bayes Classifier", In Proceedings of the IJCAI-01 Workshop on Empirical Methods in Artificial Intelligence. 2001.

[12]Irina Rish, Joseph Hellerstein, Jayram Thathachar; "An Analysis of Data Characteristics that affect Naïve Bayes Performance", IBM T.J. Watson Research Center 30 Saw Mill River Road, Hawthorne, NY 10532, USA. 2001.

[13]Pedro Domingos, Michael Pazzani; "On the Optimality of the Simple Bayesian Classifier under Zero-One Loss, Machine Learning", Vol. 29, No. 2-3, pp.103-130. 1997.
[14]Sang-Bum Kim, Hue-Chang Rim, Dong-Suk Yook, Huei-Seok Lim; "Effective Methods for Improving Naïve Bayes Text Classification", 7th Pacific Rim International Conference on Artificial Intelligence, Vol. 2417. 2002.

[15]Susana Eyheramendy, Alexander Genkin, Wen-Hua Ju, David D. Lewis, and David Madigan; "Sparce Bayesian Classifiers for Text Categorization", Department of Statistics, RutgersUniversity.2003.

[16]Dino Isa, Lam Hong lee, V. P Kallimani, R. RajKumar, " Text Documents Preprocessing with the Bahes Formula for Classification using the Support vector machine", IEEE, Traction of Knowledge and Data Engineering, Vol-20, N0- 9 pp-1264-1272, 2008.

[17]Dino Isa,, V. P Kallimani Lam Hong lee, "Using Self Organizing Map for Clustering of Text Documents", ", Elsever, Expert System with Applications-2008.

[18]Medhat, Walaa, Ahmed Hassan, and Hoda Korashy. "Sentiment analysis algorithms and applications: A survey" Ain Shams Engineering Journal 5.4 :1093-1113, 2014.

[19]https://www.researchgate.net/publication/320250187_A_survey_of_sentiment_analysis_techniques

[20]http://www.textanalyticsworld.com/pdf/Future_directions.pdf

[21]https://www.analyticbridge.datasciencecentral.com/group/socialnetworkanalytics/forum/topics/what-are-the-main-4-challenges