THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

UML501

MACHINE LEARNING

PROJECT REPORT

ON

PRODUCT REVIEW USING SENTIMENT ANALYSIS

SUBMITTED BY:

SEHAJBIR SINGH	101603309
SHERVIL GUPTA	101603312
SHIVAM MITTAL	101603316
SIDDHANT JAIN	101603330

SUBMITTED TO:

Dr. Maninder Kaur

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SEMESTER-V

BATCH 2016-2020

B.E. Computer Engineering

THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY

Abstract:

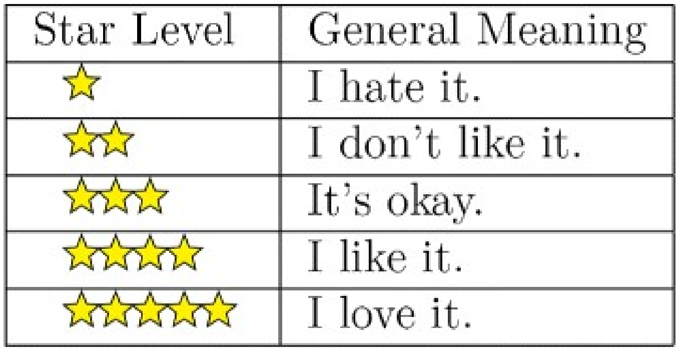
Sentiment analysis or opinion mining is one of the major tasks of NLP (Natural Language Processing). Sentiment analysis has gain much attention in recent years. Here we propose an advanced Sentiment Analysis for Product Rating system that detects hidden sentiments in comments and rates the product accordingly. The system uses sentiment analysis methodology in order to achieve desired functionality. Data used in this study are online product reviews collected from Amazon.com. We use a database of sentiment based keywords along with positivity or negativity weight in database and then based on these sentiment keywords mined in user comment is ranked. Comment are analyzed by comparing the comment with the keywords stored in database. Experiments for both sentence-level categorization and review-level categorization are performed with promising outcomes. The System takes comments of various users; based on the comment the system will specify whether the product is good or bad. The role of the admin is to add product to the system and to add keywords in database. User can easily find out correct product for his/her usage. At last, we also give insight into our future work on sentiment analysis.

1. Introduction:

Sentiment is an attitude, thought, or judgment prompted by feeling. Sentiment analysis, which is also known as opinion mining, studies people's sentiments towards certain entities. Internet is a resourceful place with respect to sentiment information. From a user's perspective, people are able to post their own content through various social media, such as forums, micro-blogs, or online social networking sites. From a researcher's perspective, many social media sites release their application programming interfaces (APIs), prompting data collection and analysis by researchers and developers. For instance, Twitter currently has three different versions of APIs available, namely the REST API, the Search API, and the Streaming API. With the REST API, developers are able to gather status data and user information; the Search API allows developers to query specific Twitter content, whereas the Streaming API is able to collect Twitter content in real-time. Moreover, developers can mix those APIs to create their own applications. Hence, sentiment analysis seems having a strong fundament with the support of massive online data.

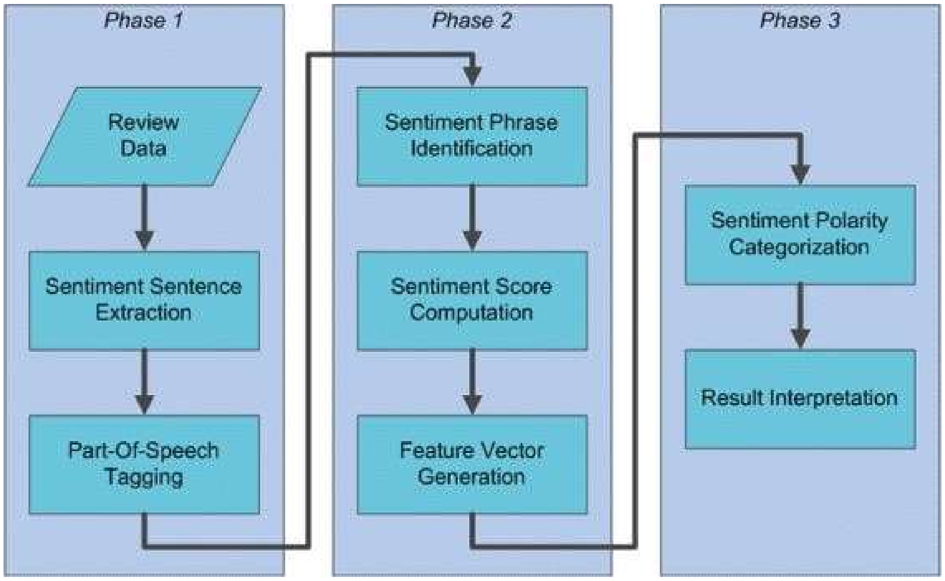
However, those types of online data have several flaws that potentially hinder the process of sentiment analysis. The first flaw is that since people can freely post their own content, the quality of their opinions cannot be guaranteed. For example, instead of sharing topic-related opinions, online spammers post spam on forums. Some spam are meaningless at all, while others have irrelevant opinions also known as fake opinions. The second flaw is that ground truth of such online data is not always available. A ground truth is more like a tag of a certain opinion, indicating whether the opinion is positive, negative, or neutral. The Stanford Sentiment 140 Tweet Corpus is one of the datasets that has ground truth and is also public available. The corpus contains 1.6 million machine-tagged Twitter messages. Each message is tagged based on the emoticons discovered inside the message.

Data used in this paper is a set of product reviews collected from Amazon. The aforementioned flaws have been somewhat overcome in the following two ways: First, each product review receives inspections before it can be posted. Second, each review must have a rating on it that can be used as the ground truth. The rating is based on a star-scaled system, where the highest rating has 5 stars and the lowest rating has only 1 star.



Star Level	General Meaning
☆	I hate it.
☆☆	I don't like it.
☆☆☆	It's okay.
☆☆☆☆	I like it.
☆☆☆☆☆	I love it.

This paper tackles a fundamental problem of sentiment analysis, namely sentiment polarity categorization [15-21]. Figure 2 is a flowchart that depicts our proposed process for categorization as well as the outline of this paper. Our contributions mainly fall into Phase 2 and 3. In Phase 2: 1) An algorithm is proposed and implemented for negation phrases identification; 2) A mathematical approach is proposed for sentiment score computation; 3) A feature vector generation method is presented for sentiment polarity categorization. In Phase 3: 1) Two sentiment polarity categorization experiments are respectively performed based on sentence level and review level; 2) Performance of three classification models are evaluated and compared based on their experimental results.


Phase 1
Review Data
Sentiment Sentence Extraction
Part-Of-Speech Tagging
Phase 2
Sentiment Phrase Identification
Sentiment Score Computation
Feature Vector Generation
Phase 3
Sentiment Polarity Categorization
Result Interpretation

1.1. Need of thesystem:

In this age of booming e-commerce, customers rarely venture to supermarkets in search of their providence, ordering stuff from websites instead. The comment/review section as well as theratings system is a boon for us consumers to skim through the vast array of assorted goods in order to find which product is good for us. However no one has time or the resources to go through all the ratings and reviews for each and every type of product for a given domain for finalising the product. We developed a sentiment analysis model which analyses the dataset, processes it and then separates it on the (1) basis of domain (2) basis of attribute. The ratings and types of comments are taken into account which later play role in determining the percentage of people who rated products positively or negatively. This model is very useful in day to day life of millennials such as ourselves who can make use of this system for a quick review of the product. Therefore the system can be used extensively for time management.

1.2. Application of proposedsystem:

- Consumer can easily decide on the product.
- Time wasted on reading multiple reviews can be better utilised.
- When applied to social media channels, it can be used to identify spikes in sentiment, thereby allowing you to identify potential product advocates or social media influencers.
- It can be used to identify when potential negative threads are emerging online regarding your business, thereby allowing you to be proactive in dealing with it more quickly.

1.3. Challenges indevelopment:

Developing the sentiment analysis model requires in depth knowledge of Natural Language processing(NLP)and its various tools such as NLTK in addition to python libraries such as pandas, numpy etc. Intricate details such as usage of stopwords had to be considered carefully. There was a problem of shaping the dataset according to the parameters of library as well. The main challenge we encountered was in inculcating the usage of TF-IDF model in sentiment analysis.

2. Existing work:

The history of natural language processing generally started in the 1950s, although work can be found from earlier periods. In 1950, Alan Turing published an article titled "Intelligence" which proposed what is now called the Turing test as a criterion of intelligence. Recent research has increasingly focused on unsupervised and semi-supervised learning algorithms. Such calculations can gain from information that has not been hand-commented on with the coveted answers, or utilizing a mix of clarified and non-explained information. By and large, this errand is substantially more troublesome than regulated learning, and normally delivers less precise outcomes for a given measure of info information.

A large number of the prominent early victories happened in the field of machine interpretation, due particularly to work at IBM Research, where progressively more muddled factual models wereproduced.

In the 2010s, representation learning and deep neural network-style machine learning methods became widespread in natural language processing, due in part to a flurry of results showing that such techniques can achieve state-of-the-art results in many natural language tasks, for example in language modelling, parsing, and many others. Popular techniques include the use of word embedding to capture semantic properties of words, and an increase in end-to-end learning of a higher-level task (e.g., question answering) instead of relying on a pipeline of separate intermediate tasks (e.g., part-of-speech tagging and dependency parsing).

Some notably successful natural language processing systems developed in the 1960s were SHRDLU, a natural language system working in restricted "blocks worlds" with restricted vocabularies, and ELIZA, a simulation of a Rogerian psychotherapist, written by Joseph Weizenbaum between 1964 and 1966. Using almost no information about human thought or emotion, ELIZA sometimes provided a startlingly human-like interaction. When the "patient" exceeded the very small knowledge base, ELIZA might provide a generic response, for example, responding to "My head hurts" with "Why do you say your head hurts?"

3. Working of the proposed system:

To achieve this project we had to follow a lot of steps.

Firstly, for working with the Tf-Idf model the dataset is required to be cleaned. All extra spaces, exclamation marks and other special characters not required are removed and a clean dataset is created.

This is followed by the application of our Tf-Idf model to separate the devices into two categories of good or bad depending on the reviews in the dataset.

Then comes the prediction part on the dataset and this is achieved using the logistic regression as it was best suited in this scenario.

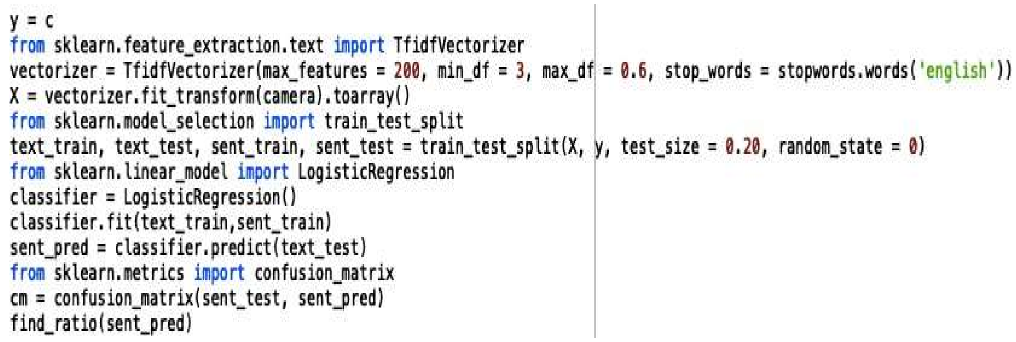
3.1. Approach followed in proposed System with discussions on Machine Learning techniques used in System

NLTK:

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum. NLTK provides a practical introduction to programming for language processing.

TF-idf

Tf-idf stands for *term frequency-inverse document frequency*, and the tf-idf weight is a weight often used in information retrieval and text mining. This weight is a statistical measure used to evaluate how important a word is to a document in a collection or corpus. The importance increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus. Variations of the tf-idf weighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query. One of the simplest ranking functions is computed by summing the tf-idf for each query term; many more sophisticated ranking functions are variants of this simple model. Tf-idf can be successfully used for stop-words filtering in various subject fields including text summarization and classification.



```
y = c
from sklearn.feature_extraction.text import TfidfVectorizer
vectorizer = TfidfVectorizer(max_features = 200, min_df = 3, max_df = 0.6, stop_words = stopwords.words('english'))
X = vectorizer.fit_transform(camera).toarray()
from sklearn.model_selection import train_test_split
text_train, text_test, sent_train, sent_test = train_test_split(X, y, test_size = 0.20, random_state = 0)
from sklearn.linear_model import LogisticRegression
classifier = LogisticRegression()
classifier.fit(text_train,sent_train)
sent_pred = classifier.predict(text_test)
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(sent_test, sent_pred)
find_ratio(sent_pred)
```

4. Data Collection and Data Preparation, Cleaning of data, Preprocessing:

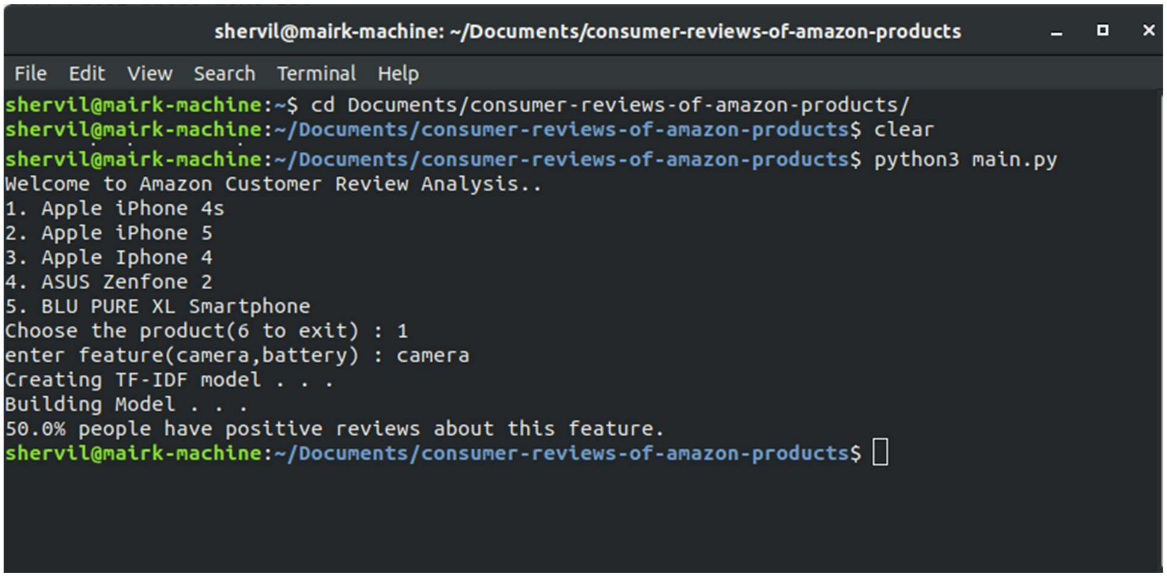
We have taken our dataset of more than 20000 reviews from kaggle. The dataset when imported, contains three columns which are Name of the mobile phone, Rating and Reviews given by various customers. Afterwards, the data is cleansed with the help of natural language processing so to remove extra keywords that were present in it. At the end, the data was pre-processed to set aside the unimportant features to optimize the model.

5. Training of the model:

We have trained our model the Tf-idf model which is short for **term frequency–inverse document frequency.** This is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in searches of information retrieval, text mining, and user modelling. The Tf– idf esteem expands relatively to the occasions a word shows up in the report and is balanced by the quantity of archives in the corpus that contain the word, which modifies for the way that a few words seem all the more much of the time when all is said in done. Tf– idf is a standout amongst the most famous term-weighting plans today; 83% of content based recommender frameworks in computerized libraries utilize Tf– idf.

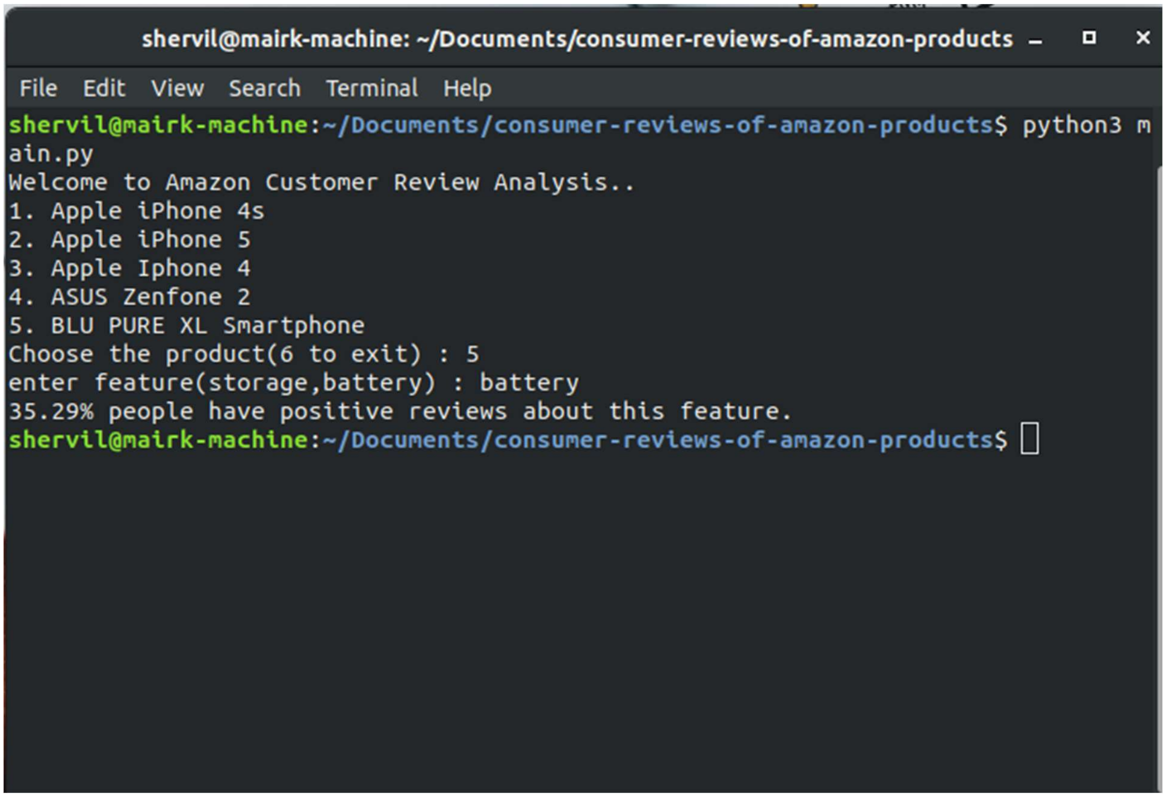
Variations of the tf–idf weighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query. Tf–idf can be successfully used for stop-words filtering in various subject fields, including text summarization and classification.

6. Testing of themodel:



shervil@mairk-machine: ~/Documents/consumer-reviews-of-amazon-products
File Edit View Search Terminal Help

```
shervil@mairk-machine:~$ cd Documents/consumer-reviews-of-amazon-products/
shervil@mairk-machine:~/Documents/consumer-reviews-of-amazon-products$ clear
shervil@mairk-machine:~/Documents/consumer-reviews-of-amazon-products$ python3 main.py
Welcome to Amazon Customer Review Analysis..
1. Apple iPhone 4s
2. Apple iPhone 5
3. Apple Iphone 4
4. ASUS Zenfone 2
5. BLU PURE XL Smartphone
Choose the product(6 to exit) : 1
enter feature(camera,battery) : camera
Creating TF-IDF model . . .
Building Model . . .
50.0% people have positive reviews about this feature.
shervil@mairk-machine:~/Documents/consumer-reviews-of-amazon-products$ 
```



shervil@mairk-machine: ~/Documents/consumer-reviews-of-amazon-products
File Edit View Search Terminal Help

```
shervil@mairk-machine:~/Documents/consumer-reviews-of-amazon-products$ python3 m
ain.py
Welcome to Amazon Customer Review Analysis..
1. Apple iPhone 4s
2. Apple iPhone 5
3. Apple Iphone 4
4. ASUS Zenfone 2
5. BLU PURE XL Smartphone
Choose the product(6 to exit) : 5
enter feature(storage,battery) : battery
35.29% people have positive reviews about this feature.
shervil@mairk-machine:~/Documents/consumer-reviews-of-amazon-products$ 
```

7. Result and discussion:

The sentiment analysis model preprocesses the dataset and rewrites it into readable format. Then the dataset is split in accordance with the product category and later attribute. The model analyses the data and ratings by matching keywords and then performing binary classification where 0 means bad and 1 means good. The model calculates and returns the percentage of people who rated and reviewed the product as good.

8. Conclusion and futurescope:

Sentiment analysis or opinion mining is a field of study that analyzes people's sentiments, attitudes, or emotions towards certain entities. This paper tackles a fundamental problem of sentiment analysis, sentiment categorization. Online product reviews from Amazon.com are selected as data used for this study. A sentiment binary categorization process has been proposed along with detailed descriptions of each step. Experiments for both sentence-level categorization and review-level categorization have been performed. The scope of the product can be further increased by adding a user friendly U/I interface. The system can be made more efficient by applying different models.

References:

- https://www.wikipedia.org/
- https://www.kaggle.com/
- https://www.nltk.org/