

Scoring Reviews Using Sentiment Analysis

A project report submitted in partial fulfillment of the requirements for the award of degree of

Bachelor of Technology
in
Computer Science and Engineering

By

A.U.N.V Maheswari [N120401]

K. Mani [N120642]

Under the Supervision of

Mr. R.Upendar Rao



Department of Computer Science and Engineering
Rajiv Gandhi University of Knowledge Technologies – Nuzvid
Nuzvid, Krishna – 521202

April, 2018



RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

(A.P. Government Act 18 of 2008)
RGUKT-NUZVID, Krishna Dist - 521202
Tele Fax : 08656 – 235557/235150

CERTIFICATE OF COMPLETION

This is to certify that the work entitled, “**Scoring Reviews Using Sentiment Analysis**” is the bonafied work of *A.U.N.V Maheswari, ID No: N120401* , *K.Mani , ID No: N120642* carried out under my guidance and supervision for the partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology** in the department of Computer Science and Engineering under RGUKT IIIT Nuzvid. This work is done during the academic session August 2017 – April 2018, under our guidance.

Mr.R.Upendar Rao

Project Supervisor
Assistant prof Dept. of CSE
RGUKT IIIT Nuzvid, Nuzvid

Mr. R.Upendar Rao

Head of Department
Assistant Professor Dept. of CSE
RGUKT IIIT Nuzvid, Nuzvid



RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

(A.P. Government Act 18 of 2008)
RGUKT-NUZVID, Krishna Dist - 521202
Tele Fax : 08656 – 235557/235150

CERTIFICATE OF EXAMINATION

This is to certify that the work entitled, “**Scoring Reviews Using Sentiment Analysis**” is the bonafied work of *A.U.N.V Maheswari , ID No: N120401, K.Mani ,ID No: N120642* and here by accord our approval of it as a study carried out and presented in a manner required for its acceptance in the partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology** for which it has been submitted. This approval does not necessarily endorse or accept every statement made, opinion expressed or conclusion drawn, as a recorded in this thesis. It only signifies the acceptance of this thesis for the purpose for which it has been submitted.

Mr. R.Upendar Rao

Project Supervisor
Assistant Prof Dept. of CSE
RGUKT IIIT Nuzvid, Nuzvid

Examiner

Project Examiner
Assistant Professor Dept. of CSE
RGUKT IIIT Nuzvid, Nuzvid



RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

(A.P. Government Act 18 of 2008)
RGUKT-NUZVID, Krishna Dist - 521202
Tele Fax : 08656 – 235557/235150

DECLARATION

We, *A.U.N.V Maheswari, ID No: N120401, K.Mani, ID No: N120642*, hereby declare that the project report entitle “**Scoring Reviews Using Sentiment Analysis**” done by us under the guidance of **Mr. R.Upendar** Rao is submitted for the partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology in Computer Science and Engineering** the academic session August 2017 – April 2018 at RGUKT – Nuzvid.

We also declare that this project is a result of our own effort and has not been copied or imitated from any source. Citations from any websites are mentioned in the references.

The results embodied in this project report have not been submitted to any other university or institute for the award of any degree or diploma.

Date :

A.U.N.V Maheswari [N120401]

Place : Nuzvid

K.Mani [N120642]

ACKNOWLEDGEMENT

We would like to express our profound gratitude and deep regards to my guide **Mr. R.Upendrar Rao** for his exemplary guidance, monitoring and constant encouragement throughout the course of this thesis.

We are extremely grateful for the confidence bestowed in us and entrusting our project entitled “Best Rating to a Product”.

At this juncture we feel deeply honored in expressing our sincere thanks to him for making the resources available at right time and providing valuable insights leading to the successful completion of our project.

We would like to thank RGUKT Nuzvid Director, faculty and staff for their valuable suggestions and discussions.

Last but not least I thank almighty, and I place a deep sense of gratitude to my family members and my friends who have been constant source of information during the preparation of this project work.

A.U.N.V Maheswari

K. Mani

ABSTRACT

The main purpose of this Project is to reduce the time of customer for reading reviews when he/she wants to buy a product. Now a days every one busy with their works so they don't have enough time to read each and every review for a particular product (product may be anything ex: books , movie etc). So that here we are providing rating to a particular product by considering all the reviews of that product from online shopping websites. So customer can choose products based on rating only , without going to all the reviews of that products. Here we are developing the website which contains product categories and products. Whenever user login into the website he can see all the product categories .If he/ she select any particular category then he can see the all products which are belongs to that particular category. Below the products there is link called View Details whenever user click on that link then user will get details of particular product. Based on that product rating user will take decision whether he/she to buy a product or not. Because of giving rating to the product we are reducing customer time without reading all the reviews in order to buy a new product.

CONTENTS

| | |
|---|-----|
| Certificate of completion | ii |
| Certificate of Examination | iii |
| Declaration | iv |
| Acknowledgement | v |
| Abstract | vi |
| Abbreviations | vii |
| Introduction | 01 |
| 1.1. Technologies Preferred | 02 |
| 1.1.1 HTML..... | 02 |
| 1.1.2 PHP..... | 03 |
| 1.1.3 MySQL | 04 |
| 1.1.4 PYTHON | 05 |
| 1.1.5 NLTK | 06 |
| 1.2 System Architecture | 06 |
| 1.2.1 Phase 1..... | 07 |
| 1.2.2 Phase 2..... | 07 |
| 1.2.3 Phase 3..... | 08 |
| 2. Literature Review | 09 |
| 2.1. Praveen Raja..... | 09 |
| 2.2. Pappu Rajan..... | 09 |
| 2.3. Arti Buche..... | 09 |
| 2.4. Sindhu..... | 09 |
| 2.5. Jisha Manjaly..... | 10 |
| 2.6. Aman Asmi..... | 10 |
| 2.7. Hassan Saif..... | 10 |

| | |
|--|-----------|
| 2.8. Praveen Raja..... | 10 |
| 2.9. Pappu Rajan..... | 10 |
| 2.10. Arti Buche..... | 10 |
| 3. Our Implementation and Experimental Results..... | 11 |
| 3.1. UML Diagrams..... | 11 |
| 3.1.1 Class Diagram..... | 11 |
| 3.1.2 Object Diagram..... | 12 |
| 3.1.3 Use Case Diagram | 13 |
| 3.1.4 Activity Digram | 14 |
| 3.1.5 Sequence Diagram | 15 |
| 3.2. Project Screenshots..... | 20 |
| 3.3. Project Code | 26 |
| 4. Conclusion and Future Scope..... | 27 |
| References..... | 28 |

ABBREVIATIONS

| | |
|------|----------------------------|
| PHP | Hypertext Preprocessor |
| SQL | Structured Query Language |
| NLTK | Natural language Tool Kit |
| HTML | Hyper Text Markup Language |
| CSS | Cascading Style Sheet |

CHAPTER 1

Introduction:

Technologies

- HTML
- PHP
- MYSQL
- PYTHON
- NLTK

System Architecture

- Phase 1
 - Review Data
 - Tokenization
- Phase 2
 - Sentiment Sentence Extraction
 - Parts Of Speech Tagging
- Phase 3
 - Sentiment Polarity Categorization
 - Result Interpretation

CHAPTER 1

INTRODUCTION

With the recent proliferation of online shopping, customer usually publish reviews to the store and product after online shopping. At the same time, the online reviews become an important approach for the potential customer to know about the store and product. They usually check the online reviews to make decision whether buy the product or not. However, the reviews grow rapidly, which make it difficult for the potential customer to read the reviews carefully one by one. It also makes it difficult for the seller to keep track and understand customer sentiment. Consequently, sentiment analysis has become a popular topic of many researchers.

The aim of sentiment analysis is to find out the positive and negative feelings, emotions and opinions written in a text. These sentiments are based on the meanings of words used in text according to different scenarios and situations. There are a variety of ways used to express the same feeling in a written text by using different grammatical rules. These grammatical rules contains negations that are very frequently used in text that completely change the meanings of words. In other words negation identification is an important aspect of sentiment analysis, it is yet to be properly addressed. In general, the efforts put into sentiment analysis of sentences having negation terms in them are less efficient with respect to general sentiment analysis. Negation identification is not a simple task and its complexity increases, since negation words such as not, nor etc., (syntactic negation) are not the only criterion for negation calculation. The linguistic patterns-prefixes (e.g., un-, dis-, etc.) or suffixes (e.g., -less) also introduce the context of negation in textual data.

The objective of this paper is to improve the performance of sentence level sentiment analysis. Here we are using Dictionary mapping to give polarity to a words. Dictionary Mapping will take constant time to map with the dictionary words with sentiment words.

1.1 TECHNOLOGIES PREFERRED

We prefer some technologies and frameworks for this project. Those are HTML,CSS for user interface, PHP for dynamic website and MYSQL for Database.

1.1.1 HTML

Hypertext Markup Language (HTML) is the standard **markup language** for creating web pages and web applications. With **Cascading Style Sheets** (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web.

Use of HTML

- HTML is used to describe the structure of pages displayed in web browsers. It is not technically a programming language, but a descriptive structural language whose origins date back to the SGML (Standard Generalized Markup Language) developed around 1986 at The European Organization for Nuclear Research (CERN) and standardized by the International Organization for Standardization (ISO) as ISO 8879.
- The first versions of HTML were based on the SGML standard, but later deviated to emulate the strictly-defined structure of EXtensible Markup Language (XML), as found in the XHTML 1.0 standards and the XHTML 1.1 drafts.
- With the introduction of HTML 5 in 2014, the bonds to XML have been loosened in favor of more semantic code which more clearly represented the structure of the documents it contained; many developers, however, still observe XML-based syntax in their HTML 5 code because of its clarity.
- **Browser Support:** It is supported by all popular browsers. Ex: Mozilla Firefox, Google Chrome, Internet Explorer etc.
- **Easy to get started:** With just the knowledge of HTML and CSS anyone can get started the browser..

1.1.2 PHP

PHP (Hypertext Preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML. The goal of the language is to allow web developers to write dynamically generated pages quickly. PHP is an amazing and popular language!

- It is powerful enough to be at the core of the biggest blogging system on the web (WordPress)!
- It is deep enough to run the largest social network (Facebook)! It is also easy enough to be a beginner's first server side language!

So what makes PHP 7 so special? What does this mean for a developer?

1. **Speed:** The developers worked very hard to refactor the PHP codebase in order to reduce memory consumption and increase performance. And they certainly succeeded.

2. **Type Declarations:** Type declarations simply means specifying which type of variable is being set instead of allowing PHP to set this automatically. PHP is considered to be a weak typed language. In essence, this means that PHP does not require to declare data types. Variables still have data types associated with them but one can do radical things like adding a string to an integer without resulting in an error. Type declarations can help define what should occur so as to get the expected results.

3. **Error Handling:** Handling fatal errors in the past has been next to impossible in PHP. A fatal error would not invoke the error handler and would simply stop script. On a production server, this usually means showing a blank white screen, which confuses the user and causes credibility to drop. It can also cause issues with resources that were never closed properly and are still in use or even locked. In PHP 7, an exception will be thrown when a fatal and recoverable error occurs, rather than just stopping the script. Fatal errors still exist for certain conditions, such as running out of memory, and still behave as before by immediately stopping the script.

4. **New Operators:** PHP 7 also brings some new operators like The spaceship operator, or Combined Comparison Operator and the Null Coalesce Operator.

5. **Easy User-land CSPRNG:** User-land refers to an application space that is external to the kernel and is protected by privilege separation, API for an easy to use and Cryptographically Secure Pseudo Random Number Generator in PHP.

1.1.3 MYSQL

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. its consistent fast performance, high reliability and ease of use. It's used in more than 6 million installations ranging from large corporations to specialized embedded applications on every

continent in the world. MySQL runs on more than 20 platforms. It is the world's most popular open source database. With its proven performance, reliability and ease-of-use, MySQL has become the leading database choice for web-based applications, used by high profile web properties including Facebook, Twitter, YouTube, Yahoo! and many more.

Oracle drives MySQL innovation, delivering new capabilities to power next generation web, cloud, mobile and embedded applications. The MySQL Database Server is very fast, reliable, scalable, and easy to use. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

Advantages of Using MySQL

- 1. Easy to use:** While a basic knowledge of SQL is required—and most relational databases require the same knowledge—MySQL is very easy to use. With only a few simple SQL statements, you can build and interact with MySQL.
- 2. Secure:** MySQL includes solid data security layers that protect sensitive data from intruders. Rights can be set to allow some or all privileges to individuals. Passwords are encrypted.
- 3. Inexpensive:** MySQL is included for free with NetWare® 6.5 and available by free download from MySQL Web site.
- 4. Fast:** In the interest of speed, MySQL designers made the decision to offer fewer features than other major database competitors, such as Sybase* and Oracle*. However, despite having fewer features than the other commercial database products, MySQL still offers all of the features required by most database developers.
- 5. Scalable:** MySQL can handle almost any amount of data, up to as much as 50 million rows or more. The default file size limit is about 4 GB. However, you can increase this number to a theoretical limit of 8 TB of data.
- 6. Manages memory very well:** MySQL server has been thoroughly tested to prevent memory leaks.
- 7. Runs on many operating systems:** MySQL runs on many operating systems, including Novell NetWare, Windows* Linux*, many varieties of UNIX* (such as Sun* Solaris*, AIX, and DEC* UNIX), OS/2, FreeBSD*, and others.

8. Supports several development interfaces:

Development interfaces include JDBC, ODBC, and scripting (PHP and Perl), letting you create database solutions that run not only in your NetWare 6.5 environment, but across all major platforms, including Linux, UNIX, and Windows.

9.Strong Data Protection:

Because guarding the data assets of corporations is the number one job of database professionals, MySQL offers exceptional security features that ensure absolute data protection. In terms of database authentication, MySQL provides powerful mechanisms for ensuring only authorized users have entry to the database server, with the ability to block users down to the client machine level being possible. SSH and SSL support are also provided to ensure safe and secure connections.

1.1.4 PYTHON

- Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales .
- Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.
- Python interpreters are available for many operating systems. CPython, the reference implementation of Python, is open source software[28] and has a community-based development model, as do nearly all of its variant implementations. CPython is managed by the non-profit Python Software Foundation.

1.1.5 NLTK

- The Natural Language Toolkit, or more commonly NLTK, is a suite of libraries and programs for symbolic and statistical natural language processing (NLP) for English written in the Python programming language. It was developed by Steven Bird and Edward Loper in the Department of Computer and Information Science at the University of Pennsylvania. NLTK includes graphical demonstrations and sample data. It is accompanied by a book that explains the underlying concepts behind the language processing tasks supported by the Toolkit, plus a cookbook.
- NLTK is intended to support research and teaching in NLP or closely related areas, including empirical linguistics, cognitive science, artificial intelligence, information retrieval, and machine learning. NLTK has been used successfully as a teaching tool, as an individual study tool, and as a platform for prototyping and building research systems. There are 32 universities in the US and 25 countries using NLTK in their courses. NLTK supports classification, tokenization, stemming, tagging, parsing and semantic reasoning functionalities.

1.2 System Architecture:

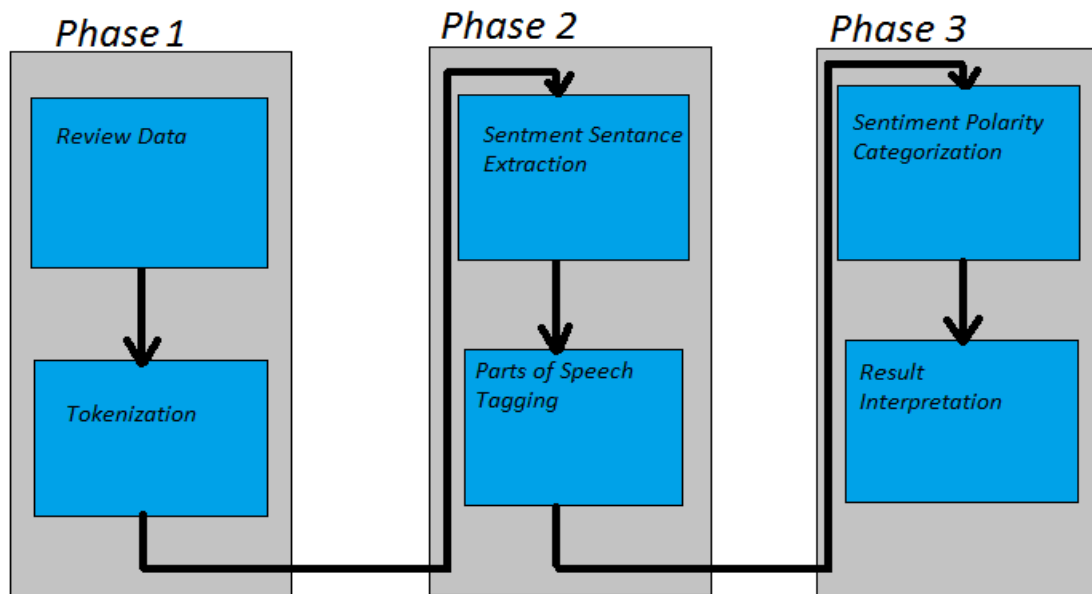


Figure 1.2: Design Phases of Giving Rating to a Product

Description:

Phase 1

Review Data

Collecting Data From Any Online Shopping Websites for any Products.

Tokenization

Tokenization is the act of breaking up a sequence of strings into pieces such as words, keywords, phrases, symbols and other elements called tokens. Tokens can be individual words, phrases or even whole sentences. In the process of tokenization, some characters like punctuation marks are discarded.

Phase 2

Sentiment Sentence Extraction

Sentiment Sentence Extraction Means Extracting the sentences which give sentiment. It is nothing but removing stop words. Stopwords are words which does not give any sentiment.

Stop Words

- A stop word is a commonly used word (such as “the”, “a”, “an”, “in”) that a search engine has been programmed to ignore, both when indexing entries for searching and when retrieving them as the result of a search query.
- We would not want these words taking up space in our database, or taking up valuable processing time. For this, we can remove them easily, by storing a list of words that you consider to be stop words. NLTK(Natural Language Toolkit) in python has a list of stopwords stored in 16 different languages. You can find them in the nltk_data directory. home/ pratima / nltk_data /corpora/ stopwords is the directory address.(Do not forget to change your home directory name).

Parts Of Speech Tagging

- Part-of-speech tagging (POS tagging or PoS tagging or POST), also called grammatical tagging or word-category disambiguation, is the process of marking up a word in a text (corpus) as corresponding to a particular part of speech, based on both its definition and its context—i.e., its relationship with adjacent and related words in a phrase, sentence, or paragraph. A simplified form of this is commonly taught to school-age children, in the identification of words as nouns, verbs, adjectives, adverbs, etc.
- Once performed by hand, POS tagging is now done in the context of computational linguistics, using algorithms which associate discrete terms, as well as hidden parts of speech, in accordance with a set of descriptive tags. POS-tagging algorithms fall into two distinctive groups: rule-based and stochastic. E. Brill's tagger, one of the first and most widely used English POS-taggers, employs rule-based algorithms.

Phase3

Sentiment Polarity Categorization

In this Phase We are calculating the polarity of words as positive ,negative or neutral. For giving polarity to a words in sentiment sentences we are using dictionary mapping. The Use of dictionary mapping is it takes constant time for comparing the every sentiment word with the words in dictionary.

Result Interpretation

After giving polarity to every sentiment words in sentiment sentences we are calculating the sentiment of the total review by calculating the total positive words and total negative words and total sentiment words. Using positive,Negative and total sentiment words we can calculate the Rating of the review.

By using below formula.

$$\text{Recommendation Percentage} = (\text{positive_words} / \text{Total_sentimet_words}) * 100$$

CHAPTER 2

LITERATURE REVIEW

- *Praveen Raja*[1]
- *Pappu Rajan*[2]
- *Arti Buche*[3]
- *Sindhu*[4]
- *Jisha Manjaly*[5]
- *Aman Asmi*[6]
- *Hassan Saif*[7]
- *Xing Fang and Tustin Zhan*[8]
- *Doaa Mohey El-Din, Hoda M.O. Mokhtar and Osama Ismae*[9]
- *F.Morales, M.Ruiz nad L.Velasco*[10]

CHAPTER 2

LITERATURE REVIEW

2.1 Praveen Raja[1]:

This paper clearly explained that the outlook analysis is done on a micro blogging data .The words are expressed in micro blogging sites are compared with those in each other data that have been previously labeled as positive or negative .After looking at these expressions ,the algorithm then judges whether the text in the micro blogging sites and is positive or negative based on the chances for each possibility.

2.2 Pappu Rajan[2]:

The objective of this paper is to determine the sentiment of the text whether it is positive or negative which is extended to strength of polarity also this approach is used to obtain the significant feature and to analyzing the overall sentiment for each object by computing the weighted average for all the sentiments in the textual data. This paper mainly focused on twitter data which is collected from R tool.

2.3 Arti Buche[3]:

It clearly explained that the sentiment analysis is a type of Natural Language Processing which is used to track products, brands in the web. It has thus become a necessity for companies to collect data from various sources such as blogs, review sites, Micro-blogs and there-by determining whether they are viewed positively or negatively using part of speech tagging.

2.4 Sindhu[4]:

This paper explains the sentiment analysis as a computational techniques for analyzing the opinions that are extracted from various sources like the blog posts ,comments on forums, reviews about products, policies any topic on social networking sites or tweets. the process of selecting the opinionated sentences and ignoring the factual sentences is called subjectivity detection which is then pre processed by tokenizing stop words filtering and stemming.

2.5 Jisha Manjaly[5]:

This paper discussed about social media such as blogs ,twitter, face book are widely used for participatory information sharing and collaboration .the opinions are then classified to positive or negative and neutral which is further divided into the emotion states such as sad, happy and angry.

2.6 Aman Asmi[6]:

This paper explains sentiment analysis is a task of detecting, extracting and classifying opinions and sentiment expressed in texts. It includes the identifications of the meaning of words within the text through natural language processing rules. Main achievement of the paper is a demonstration on an approach for automatic identification and calculations of negation in opinion and sentiment analysis.

2.7 Hassan Saif[7]:

This paper discussed SentiCircles, a lexicon-based approach for sentiment analysis on Twitter.Polarities of words regardless of their context, SentiCircles takes into account the co-occurrence patterns of words in different contexts in tweets to capture their semantics and update their pre-assigned strength and polarity in sentiment lexicons .

2.8 Xing Fang and Tustin Zhan[8]:

This paper presents A general process for sentiment polarity categorization is proposed with detailed process description. Experiments for both sentence-level categorization and review-level categorization are performed with promising outcomes.

2.9 Doaa Mohey EI-Din,Hoda M.O. Mokhtar and Osama Ismae[9]:

This paper proposes a new technique to analyze online reviews, It is called sentiment analysis of online papers(SAOOP). SAOOP is a new technique used for enhancing bag-of –words model, improving the accuracy and performance. SAOOP is useful in increasing the understanding rate of review’s sentences through higher language coverage cases. SAOOP introduces solutions for some sentiment analysis challenges and uses them to achieve higher accuracy.

2.10 F.Morales,M.Ruiz nad L.Velasco[10]:

This paper presents A set of modules to support data analytics-based algorithms along with a machine learning procedure based on artificial neural networks(ANN) That provides robust and adaptive traffic models.

CHAPTER 3

IMPLEMENTATION AND EXPERIMENTAL RESULT

Diagrams

➤ UML Diagrams

- Class Diagram
- Object Diagram
- Use Case Diagram
- Activity Diagram
- Sequence Diagram

➤ Website Diagrams

- Website Home Page
- User Login Page
- User Registration Page
- Main Page
- Product categories
- Products
- *Product Details*

CHAPTER 3

IMPLEMENTATION AND EXPERIMENTAL RESULT

Website:

Developed Website is “Scoring Reviews Using Sentiment Analysis, which is used to provide rating to product by taking reviews from online shopping websites. It will contain Registration, Login, for User. First of all user should register by filling his credentials in registration page. After registration he had an account in our website, So by using his name and password he will login into the website. After user Login with his/her Credentials He will get Home Page and he can see many products categories .He can choose any category then he will get product which are related to selected category and select any product. After choosing product there is a link called View Details Below the product .

if user click on that Link then User will get All the Details Of that Product. Rating of that product is also there in the product Details .User can See Rating of that product and Based on the Rating User can Decided wheather to buy a product or not.

This site is used as third party to a user he can see the see the rating of the product . He can see the rating of any product he want but he can’t give reviews to a product .

3.1 UML Diagrams

3.1.1 Class Diagram:

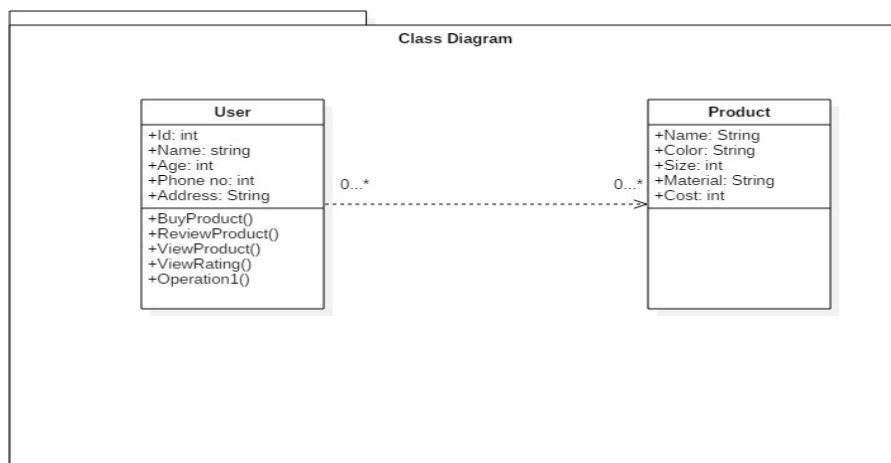


Figure 3.1.1

Figure 3.1.1 Description:

This is class diagram which represents classes. Every class contains attributes and methods which act on those classes. In our project we have two classes one is User and another one is Product. Class diagram also represents mapping cardinality between the classes. Here mapping cardinality between user and product is many to many.

3.1.2 Object Diagram:

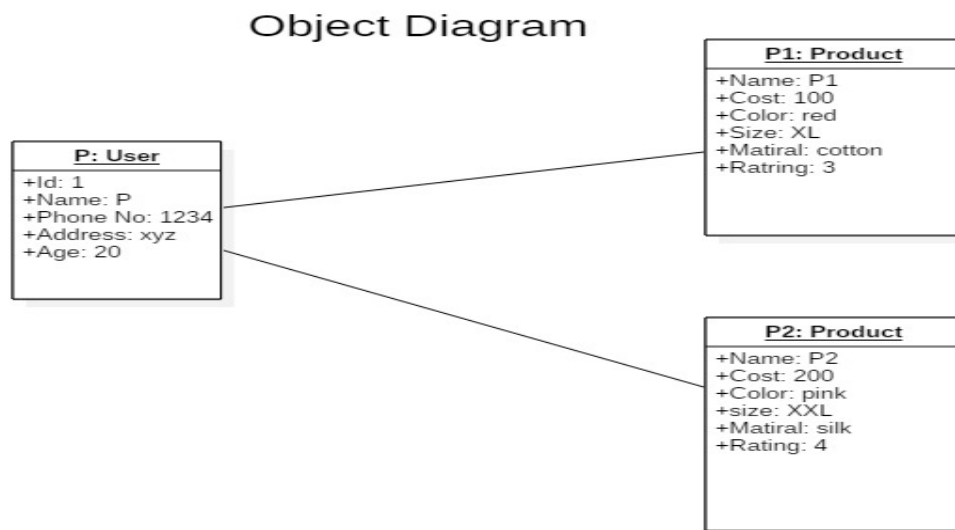


Figure 3.1.2

Figure 3.1.2 description:

This is Object diagram which represents objects. Every object have attributes and instances . Instances also have attributes, here Object is user and instances are P1:Product1 and P2: Product2.

3.1.3 Use Case Diagram:

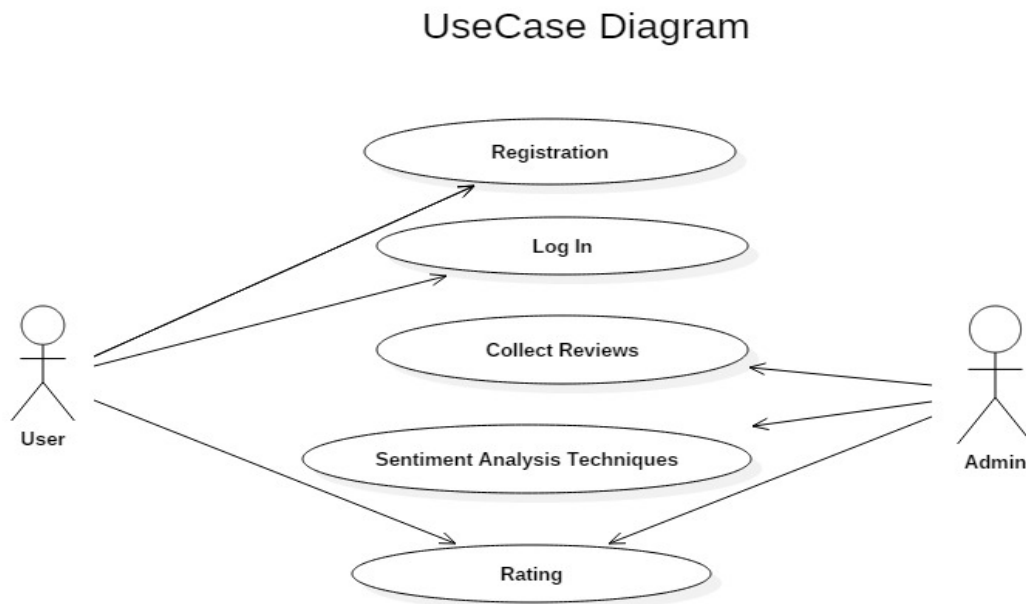


Figure 3.1.3

Figure 3.1.3 description:

This is UseCase diagram which represents which things or persons presented in the project and actions performed by each thing or person.

User actions: can register to the website and login into the website and he/she can see the rating of the product.

Admin actions: Admin can collect reviews from online shopping websites for any product and he/she can apply some sentiment analysis algorithms on the data he/she collected. After applying sentiment analysis algorithms he/she can give rating to the product.

3.1.4 Activity Diagram:

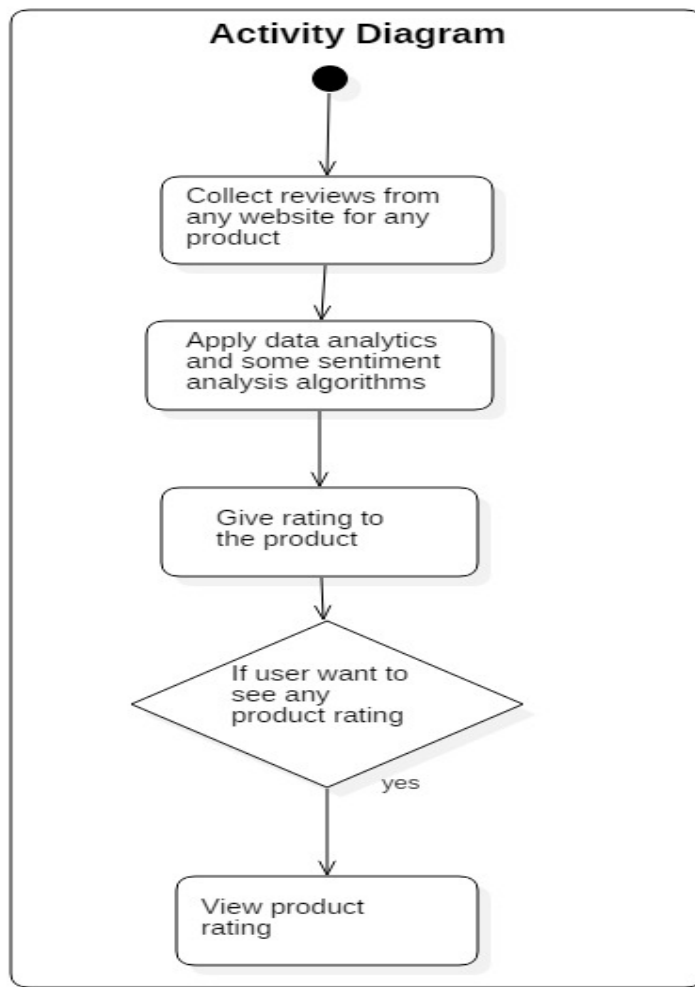


Figure 3.1.4

Figure 3.1.4 description: This diagram represents the flow of actions of this project. Here Admin collect reviews from any online shopping websites for any product after that he/she apply some data analytics and some sentiment analysis algorithms. By applying data analytics and sentiment analysis algorithms he/she can give rating to the product. If any user wants any product rating he/she can register to the website and login into the website. After login into the website he/she can see all categories of the products and he/she can choose product belongs to any category and he/she can view the rating of any selected product by clicking view details link.

3.1.5 Sequence Diagram:

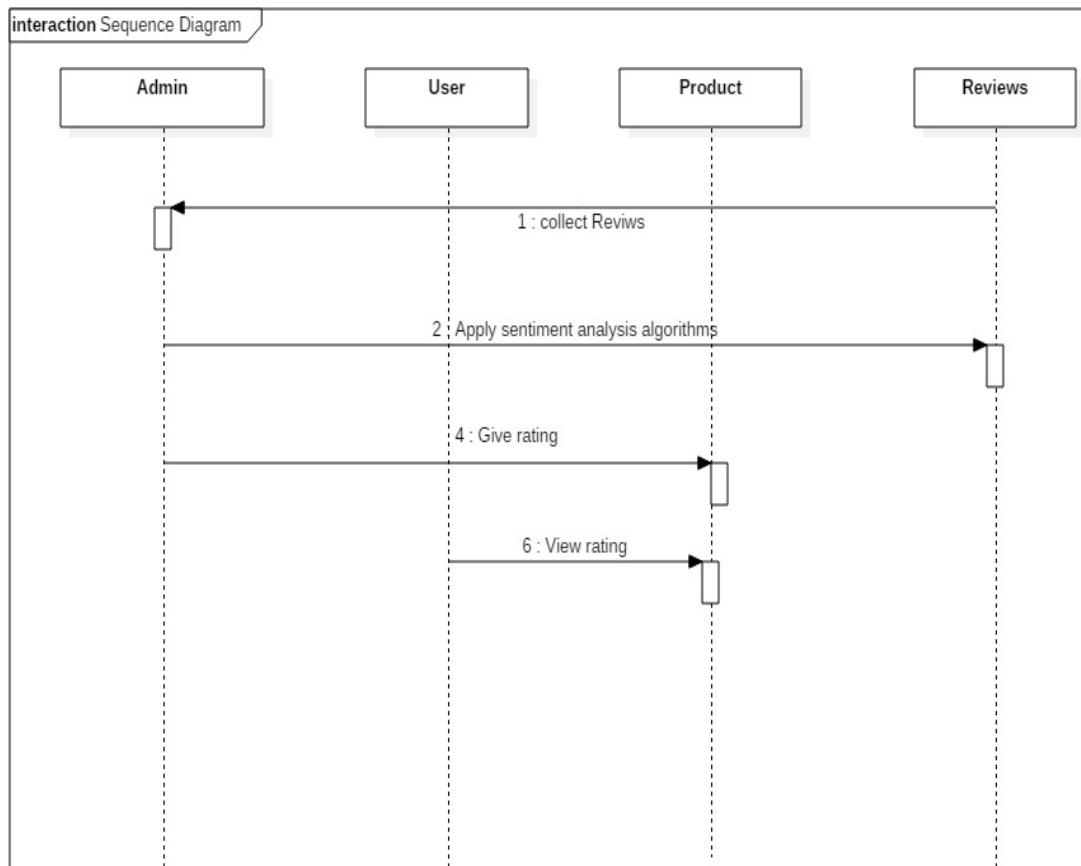


Figure 3.1.5

Figure 3.1.5 description:

This is sequence diagram which represents sequence flow of our project. We have 4 categories Admin , User , Product, Reviews . First Admin start the action Admin collect reviews from online shopping websites for any product. After that admin apply some data analytics and some sentiment analysis algorithms on collected reviews. After applying data analytics and sentiment analysis algorithms he/she can give rating to the product. If user wants to see rating to the product he can view the rating.

3.2.1 Website Home Page:



Figure:3.2.1

Description:

The above figure is our main website .whenever user open our website he/ she will get this page.

3.2.2 User Login Page:



Figure:3.2.2

Figure 3.2.2 Description:

This is user login page if user has already registered then he/she will login into this website by entering with his/her credentials. If User does not have account then he/she create account and login with his/her credentials.

3.2.3 User Registration Page:

REGISTRATION FORM

First Name

Last Name

Age

Gender ☐ Female ☐ Male

User Name

Contact Number

Password

Security Question

Security Answer

Figure:3.2.3

Description:

This is registration page user can register to our website by filling all the fields in this registration page.

3.2.4 Main website:



Figure:3.2.4

Description:

This is main webpage. Whenever user login into our site he/she will get this page.

3.2.5 Product Categories:

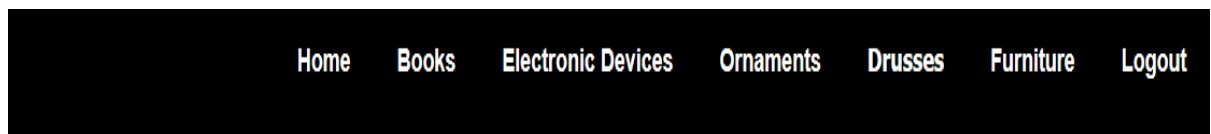


Figure:3.2.5

Description: These are categories in our website .user can view all the categories.

Books: Books categories contain all the books

Electronic Devices: This category contain all the electronic devices

Ornaments: This category contain all the ornaments

Dresses: This category contain again three sub filed's 1.for men 2. For women 3.For Children

1.For Men : This category contain all the men's wear.

2.For Women: This category contain all the women's wear.

3.For Children: This category contain all the children' s wear.

3.2.6 products:

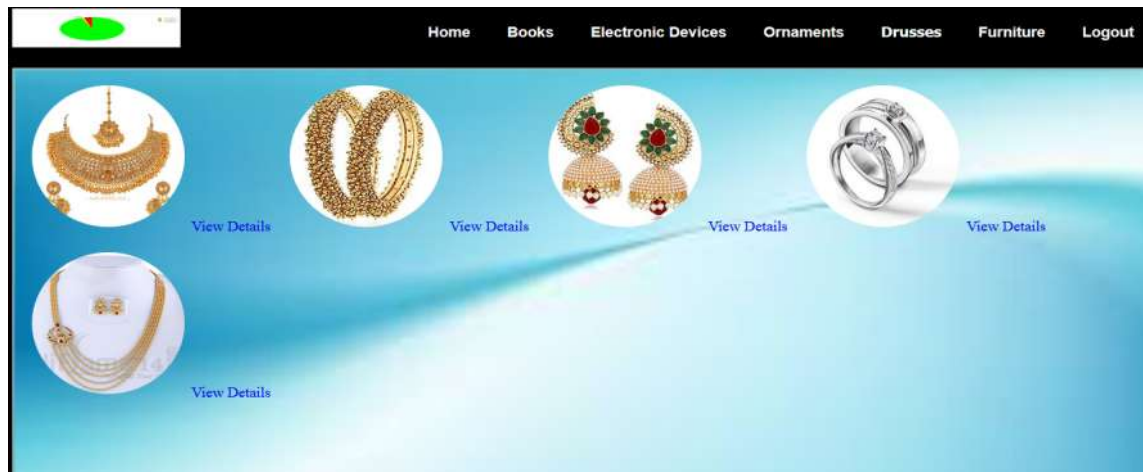


Figure:3.2.6

Description:

These are products . whenever user select any category then he/she will get the products which are present in that particular category. Here user select ornaments category so he/will get these products .

3.2.7 Product Details:

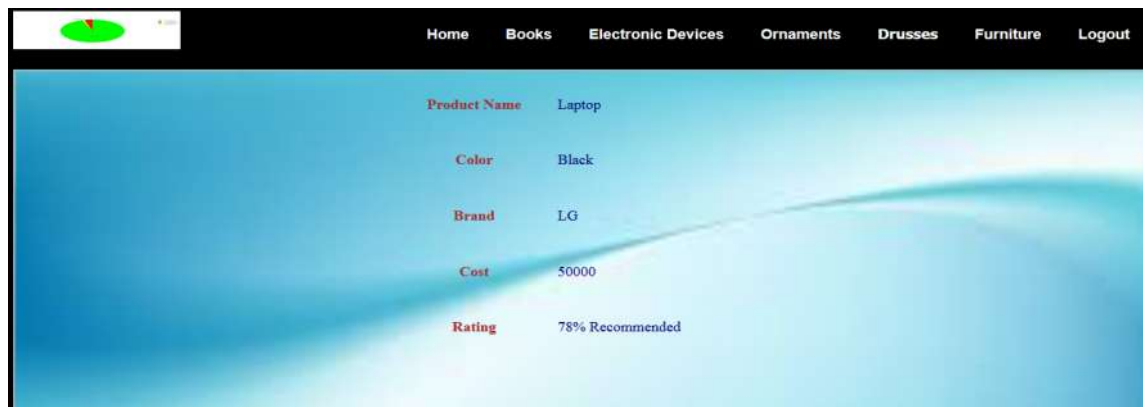


Figure:3.2.7

Description:

Below Every product there is link called view details. if user click on the link which is below any product then he/she will get the details of that particular product. User can view all the product details including product rating .Based on that product rating given by us user can decide whether to buy a product or not.

3.2.8 Product rating Output:

```
In [4]: run rer.py
('the', 'the', ['DT']),
('scrubbing', 'scrubbing', ['NN']),
('ones', 'ones', ['NNS']),
('!', '!', ['.']),
[('This', 'This', ['DT']),
 ('review', 'review', ['NN']),
 ('was', 'was', ['VBD']),
 ('collected', 'collected', ['VBN']),
 ('as', 'as', ['IN']),
 ('part', 'part', ['NN']),
 ('of', 'of', ['IN']),
 ('a', 'a', ['DT']),
 ('promotion', 'promotion', ['NN']),
 ('.', '.', ['.'])]
analyzing sentiment...

12648.0
total sentiment words...
12910.0
positive 98.98528272656854
```

Figure:3.2.8

Description:

Here analyzing sentiment means the positive words which are greater than positive words. Total sentiment words means Total number of sentiment words present in the data .Using these two we can find the total number of positive words and total number of negative words.

With positive words and negative words we give rating to a product .

The formula for Percentage of recommendation of a product:

$$((\text{Total_positive_words})/(\text{Total_sentiment_words})) * 100$$

Here we get the output as 98.98528272656854 it means Some product is 98 % recommended by our website by taking all the reviews of that product.

3.3. Project Code:

```
from pprint import pprint
```

```
import nltk
```

```
import yaml
```

```
import sys
```

```
import os
```

```
import re
```

```
class Splitter(object):
```

```
    def __init__(self):
```

```
        self.nltk_splitter = nltk.data.load('tokenizers/punkt/english.pickle')
```

```
        self.nltk_tokenizer = nltk.tokenize.TreebankWordTokenizer()
```

```
    def split(self, text):
```

```
        sentences = self.nltk_splitter.tokenize(text)
```

```
        tokenized_sentences = [self.nltk_tokenizer.tokenize(sent) for sent in sentences]
```

```
        return tokenized_sentences
```

```
class POSTagger(object):
```

```
    def __init__(self):
```

```
        pass
```

```
    def pos_tag(self, sentences):
```

```
        pos = [nltk.pos_tag(sentence) for sentence in sentences]
```

```
        #adapt format
```

```
pos = [[(word, word, [postag]) for (word, postag) in sentence] for sentence in pos]
```

```
    return pos
```

```
class DictionaryTagger(object):
```

```
    def __init__(self, dictionary_paths):
```

```
        files = [open(path, 'r') for path in dictionary_paths]
```

```
        dictionaries = [yaml.load(dict_file) for dict_file in files]
```

```
        map(lambda x: x.close(), files)
```

```
        self.dictionary = {}
```

```
        self.max_key_size = 0
```

```
        for curr_dict in dictionaries:
```

```
            for key in curr_dict:
```

```
                if key in self.dictionary:
```

```
                    self.dictionary[key].extend(curr_dict[key])
```

```
                else:
```

```
                    self.dictionary[key] = curr_dict[key]
```

```
                    self.max_key_size = max(self.max_key_size, len(key))
```

```
    def tag(self, postagged_sentences):
```

```
        return [self.tag_sentence(sentence) for sentence in postagged_sentences]
```

```
    def tag_sentence(self, sentence, tag_with_lemmas=False):
```

```
        tag_sentence = []
```

```
        N = len(sentence)
```

```
        if self.max_key_size == 0:
```

```
            self.max_key_size = N
```

```
            i = 0
```

```

while (i < N):
    j = min(i + self.max_key_size, N) #avoid overflow
    tagged = False
while (j > i):
    expression_form = ''.join([word[0] for word in sentence[i:j]]).lower()
    expression_lemma = ''.join([word[1] for word in sentence[i:j]]).lower()
    if tag_with_lemmas:
        literal = expression_lemma
    else:
        literal = expression_form
    if literal in self.dictionary:
        #self.logger.debug("found: %s" % literal)
        is_single_token = j - i == 1
        original_position = i
        i = j
        taggings = [tag for tag in self.dictionary[literal]]
        tagged_expression = (expression_form, expression_lemma, taggings)
        if is_single_token: #if the tagged literal is a single token, conserve its previous
taggings:
            original_token_tagging = sentence[original_position][2]
            tagged_expression[2].extend(original_token_tagging)
        tag_sentence.append(tagged_expression)
        tagged = True
    else:
        j = j - 1
    if not tagged:
        tag_sentence.append(sentence[i])
        i += 1

```

```

return tag_sentence

def value_of(sentiment):
    if sentiment == 'positive': return 1
    if sentiment == 'negative': return -1
    return 0

def sentence_score(sentence_tokens, previous_token, acum_score):
    if not sentence_tokens:
        return acum_score
    else:
        current_token = sentence_tokens[0]
        tags = current_token[2]
        token_score = sum([value_of(tag) for tag in tags])
        if previous_token is not None:
            previous_tags = previous_token[2]
            if 'inc' in previous_tags:
                token_score *= 2.0
            elif 'dec' in previous_tags:
                token_score /= 2.0
            elif 'inv' in previous_tags:
                token_score *= -1.0
        return sentence_score(sentence_tokens[1:], current_token, acum_score + token_score)

def sentiment_score(review):
    return sum([sentence_score(sentence, None, 0.0) for sentence in review])

def positive_count(rev):
    return sum([sentence_scores(sentence, None, 0.0) for sentence in rev])

def sentence_scores(sentence_tokens, previous_token, acum_score):

```

```

if not sentence_tokens:
    return acum_score
else:
    current_token = sentence_tokens[0]
    tags = current_token[2]
    token_score = sum([value_off(tag) for tag in tags])
    if previous_token is not None:
        previous_tags = previous_token[2]
    return sentence_scores(sentence_tokens[1:], current_token, acum_score + token_score)

def value_off(sentiment):
    if sentiment == 'positive': return 1
    if sentiment == 'negative': return -1
    return 0

if __name__ == "__main__":
    text = open("review.txt", "r").read()
    splitter = Splitter()
    postagger = POSTagger()
    dicttagger = DictionaryTagger([ 'dicts/positive.yml', 'dicts/negative.yml',
                                   'dicts/inc.yml', 'dicts/dec.yml', 'dicts/inv.yml'])
    splitted_sentences = splitter.split(text)
    pprint(splitted_sentences)
    pos_tagged_sentences = postagger.pos_tag(splitted_sentences)
    pprint(pos_tagged_sentences)
    dict_tagged_sentences = dicttagger.tag(pos_tagged_sentences)
    pprint(dict_tagged_sentences)
    print("analyzing sentiment...")
    score = sentiment_score(dict_tagged_sentences)

```

```

print(score)
print("total sentiment words...")
total_count = positive_count(dict_tagged_sentences)
print(total_count)
if score > 0:
    words = total_count-score
    positive = words/2
    negative = words/2
    total_positive_words = score+positive
    rating = (total_positive_words/total_count)*100
    print("positive",rating)
if score < 0:
    words = total_count+score
    positive = words/2
    negative = words/2
    total_negative_words = score-negative
    rating = (positive/total_count)*100
    print(rating,"positive")

```

CHAPTER 4

CONCLUSION AND FUTURE SCOPE

- Conclusion
- Future Scope

CHAPTER 4

CONCLUSION AND FUTURE SCOPE

4.1 Conclusion

The website is designed to give rating to a product by taking reviews for any product from any online shopping website. So that user can view the product Rating and based on product rating he can take decision whether to buy a product or not.

Because of giving rating to a product we can reduce the user time to reading reviews for a product. User can view product rating without reading all the reviews of that product.

4.2 Future Scope

Here we implement this website statically but in future we want to implement this website dynamically for giving rating to any product .It means in our website we give rating to limited number of products but in future we want to implement this website for more number of products and whenever user enter any product name then he/she will get the details of that product including rating also.

REFERENCES

- [1]. Praveen raja , “Web outlook analysis for attains positive or negative words using Micro Blogging and Social Networking”. Research Scholar, Department of Computer Science, IJISSET, August 2016.
- [2]. Pappu Rajan, “Web sentiment analysis for scoring positive or negative words using tweeter data,” .Research scholar, Department of Computer Science, International Journal of Computer Applications ,June 2014.
- [3]. Arti Buche,Dr.M.B.Chandak,Akshay Zadgaonkar, “Opinion Mining and Analysis: A survey,” computer science department RCOEM, June 2013.
- [4]. Sindhu, “Survey on opinion mining and sentient polarity classification” ,IJCSI International Journal of computer science Issues,Vol.9,Issue 4, 2013.
- [5]. Jisha Manjaly ,”Twitter based sentiment analysis for subject identification,” International journal of advanced research in computer and communication engineering, 2013.
- [6]. Aman Asmi ,”Negation Identification and Calculation in Sentiment Analysis,” The Second International Conference on Advances in Information Mining and Management,2012.
- [7]. Hassan Saif , “Contextual Semantics for Sentiment Analysis of Twitter,” Knowledge Media Institute, Open University, United Kingdom, june 12 2015.
- [8]. Xing Fang and Tustin Zhan, ”Sentiment Analysis using Product review Data,” Fang and Zhan Journal of Big Data,2015.
- [9]. Doaa Mohey EI-Din ,Hoda M.O. Mokhtar , Osama Ismae. “Online Paper Review Analysis,” International Journal of Advanced Computer Science Applications(IJACSA), Vol 6,No. 9, 2015.
- [10]. F.Morales,M.Ruiz nad L.Velasco. Data Analytics Based Origin-Destination Core Traffic Modling. Optical Communications Group(GCO), Universitat Politecnica de catalwnya(UPC), Barcelona,Spain.