

MINING FEATURE-OPINION FROM CUSTOMER REVIEWS FOR SENTIMENT CLASSIFICATION

Major project Submitted in partial fulfillment of the requirements

For the award of degree of

Master of Technology In Information Technology

Submitted By

KAMAL VASHISHT

(Roll No. 2K11/ISY/11)

Under the guidance of

Mr. RAHUL KATARYA

(Assistant Professor)

Department of Information Technology



Department of Information Technology

Delhi Technological University

Delhi

Session 2011-2013

CERTIFICATE

This is to certify that **Mr. Kamal Vashisht** (2k11/ISY/11) has carried out the major project titled “Mining Feature-Opinion from reviews for Sentiment Classification” as a partial requirement for the award of Master of Technology degree in Information Systems by Delhi Technological University.

The major project is a bonafide piece of work carried out and completed under my supervision and guidance during the academic session 2011-2013. The matter contained in this report has not been submitted elsewhere for the award of any other degree.

(Project Guide)

Mr. Rahul Katarya

(Assistant Professor)

Department of Information Technology

Delhi Technological University

Bawana Road, Delhi-110042

Acknowledgements

I take this opportunity to express my sincere gratitude towards **Mr. Rahul Katarya, Assistant Professor** (Information Technology) for his constant support and encouragement. His excellent guidance has been instrumental in making this project work a success.

I would like to thank **Dr. O.P. Verma**, H.O.D of Department of Information Technology for his useful insights and guidance towards the project. His suggestions and advice proved very valuable throughout.

I would like to thank members of the Department of Information Technology at Delhi Technological University for their valuable suggestions and helpful discussions.

I would also like to thank my family and friends, who have been a source of encouragement and inspiration throughout the duration of the project. I would like to thank the entire DTU family for making my stay at DTU a memorable one.

Kamal Vashisht

Roll No. 2k11/ISY/11

M.Tech (Information Systems)

E-mail: kamal1109@gmail.com

ABSTRACT

Almost all people want to gain more and more information about the products, before they purchase them. Therefore, they ask their friends, search on net and then decide to buy a product. As there is tremendous increase in e-commerce, almost every company provides a customer feedback data form on its website. Many sites put stress on participation of users, more and more Websites, such as Amazon, UCI lead people to write their opinion about products they are interested in. So, the number of product reviews from customer is also increasing. The opinion not only helps the customer to buy good product, also help the product manufacture to see the pros and cons of their product and also show the comparison of his product with the other competitor and help the product manufacturer to see which product is liked/disliked by the customer and they can improve their product future. Therefore it becomes very difficult for manufacturers to analyze every review for analyzing the product. Hence, we have made a system that takes customer reviews and finds positive and negative feature with their semantic orientation. We have used POS tagging for each sentence, and then extracted the features from the customer reviews. We have then worked on co-reference (pronoun) resolution before summarizing the semantic orientation for all features. First we have find features of the product and then reduced them using Word Net Similarity for grouping similar features. Finally, we will classify as positive, negative or neutral, so that a customer gets a better idea of each particular feature of a product and can compare with other products.

Keywords: Opinion Mining, Sentiment Orientation, Pronoun Resolution, reviews summarization, word net similarity, Text Mining.

Contents

Certificate.....	i
Acknowledgement.....	ii
Abstract.....	iii
Chapter 1: Introduction of Mining.....	1
1.1 Introduction.....	1
1.2 Introduction of Web Mining.....	1
1.3 Types of Web Mining.....	1
1.3.1 Web Usage of Mining.....	2
1.3.2 Web Content Mining.....	2
1.3.3 Web Structure mining.....	3
1.4 Need of Web Mining.....	5
1.5 Techniques and Algorithms used in web mining	5
1.5.1 Association Rule Mining.....	5
1.5.1.1 Sequence Mining.....	6
1.5.2 Supervised Learning	6
1.5.2.1 Decision Tree Induction.....	7
1.5.3 Semi-supervised Learning.....	8
1.5.3.1 Graph-based methods	8
1.5.4 Unsupervised Learning	9
Chapter 2 Opining Mining.....	10
2.1 Opining Mining	10
2.2 Need of Opinion mining.....	11
2.3 Opinion Mining Applications.....	11
2.4 Challenges in Opinion Mining.....	12
2.5 Representation of Opinion.....	13
2.5.1 Objectives of Sentiment mining.....	13
2.5.2 Types of Opinion.....	14

Chapter 3 Sentiment Mining	15
3.1 Perception of Sentiment Mining	15
3.2 Sentiment classification.....	16
3.2.1 Sentence Level Sentiment Classification.....	16
3.2.2 Document level Sentiment Classification.....	17
3.2.3 Feature Based Sentiment Classification	17
3.3 WordNet Similarity Measures.....	18
Chapter 4: Literature Survey	19
4.1 Opinion mining of Product Reviews by means of Association Rules	19
4.1.1 POS Tagging	19
4.1.2 Association Rule Mining.....	20
4.1.3 Semantic Orientation from Association	20
4.2 Product weakness from Feature based sentiment analysis	21
4.2.1 Grouping features	21
4.3 Feature and Opinion Mining	22
4.4 Mining and Summarizing Customer Reviews	22
4.4.1 The task is performed in three steps.....	23
4.4.2 Challenges	23
4.5 Examining and comparing opinion in Web.....	23
4.5.1 Two challenging task need to be implemented.....	24
4.5.2 Opinion Observer works in two stages.....	24
4.6 Clustering Product Features for Opinion Mining	25
4.6.1 Product Feature Grouping.....	25
Chapter 5: Proposed Algorithm	26
5.1 Feature Based Sentiment Mining.....	26
5.2 Proposed Architecture.....	26
5.2.1 Feature Extraction.....	27

5.2.2 Procedure	27
Chapter 6: Experimental Results.....	34
6.1 Results	34
6.2 Evaluation	50
Chapter 7: Conclusion and Future work.....	52
7.1 Conclusion	52
7.2 Future work	52
References.....	53

List of Figures

1.	Figure 1. 1 Taxonomy of Web Mining.....	2
2.	Figure 1.2 Text Mining Hierarchy.....	3
4.	Figure 5.1 Proposed Architecture of our system	27
5.	Figure 5.2 POS tagging.....	30
6.	Figure 6.1 Product Features.....	34
7.	Figure 6.2 Grouping similar features.....	35
8.	Figure 6.3 Pronoun reference.....	38
9.	Figure 6.4 Dependency relation.....	38
10.	Figure 6.5 Semantic orientation.....	40
11	Table 1 Pattern of POS tagging.....	29
12	TABLE 2 Recall and Precision rate of the system.....	51

Chapter 1

INTRODUCTION OF MINING

1.1 Introduction

Mining is basically to extract something from its source or warehouse. Data warehouse is the collection of various types of data and we can extract useful data from the data warehouse. This is called data mining [1]. Finding the precise information from such an enormous repository [2] of unstructured web information is still major part of research.

1.2 Introduction of Web Mining

Web mining can be generally defined as [3]: Extract interested, hidden patterns and useful information from the WWW resources and behaviour. There are large numbers of information available on web. Web mining is the process of extracting constructive, precious information from World Wide Web or we can say that, it is an application of data mining, to find out patterns from World Wide Web. It is the integration of information that the user needs [4].

Shiqun Yin, Yuhui Qiu, Jike Ge [5] uses the text mining technique and discusses an algorithm of how to mine as well as articulate text attribute, how we can follow the appropriate Web page or website as per the user's request, how we can classify the information of data by means of opinion judgement combined with the Web page text contents for later use.

1.3 Types of Web Mining

Web mining is divided into three types:

1. Web usage mining
2. Web content mining
3. Web structure mining

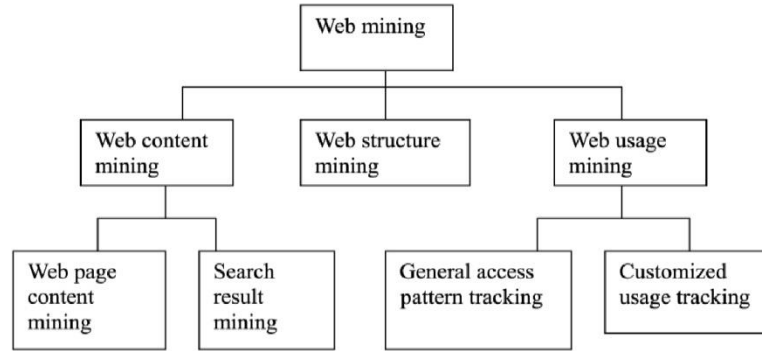


Figure 1.1 Classification of Web Mining

1.3.1 Web Usage Mining

Web Usage Mining [6, 8] is a method by which we recognize the surfing/browsing pattern by analyzing the user's navigational behavior. It keeps on tracing user's activities when the person communicates with the web.

As the name suggests that web usage mining is the process of what we search on the web. Some of the person might be searching only at multimedia data, while several others might be engrossed in textual data. Web Usage Mining is a data mining method to find out fascinating and hidden custom patterns from Web information, in order to comprehend and serve up the requirements of Web-based applications [7].

Web usage mining extracts relevant, important and implicit data from the server logs that is user's the past record. Web usage mining is the procedure [8] to locate what type of information persons are trying to find on the web. Web mining comprises of three phase, pre-processing, pattern detection and pattern examination.

1.3.2 Web Content Mining

Web content mining [9] is another category of web mining in which important and useful information is extracted from the data present on the internet. Web content mining comes under information mining and social text mining. However, it differs from information mining in the

fact that mining of web data can be semi-structured or unstructured and from text mining in the fact that text mining mainly emphasis on unstructured text [10].

It deals with finding out important, hidden sequences and facts from web page content. Margaret H. Dunham [11] states that web content mining is viewed as the extension of the work carried out by basic search engines. Web Content Mining examines the data from the web resources. Web content mining [12] can be alienated into two parts: (a) Agent based technique (b) Database based technique. Web content mining enhances the traditional search engines mining. Web content mining makes use of data mining method for scalability, performance and usefulness

1.3.3 Web Structure mining

Web structure mining, is a used to find a direct link connection or the association between Web pages interlinked by information/data. The foremost purpose for structure mining is to mine formerly unknown association between Web pages. It compacts with modeling and discovering the link structure of web. The aim of Web Structure Mining [8, 9] is to produce structured synopsis of the web page from the website. Web information recovery tool mainly uses the content available on the web pages and ignore precious, significant data enclosed in web link. Web structure mining uses tree-like arrangement to evaluate and describe the HTML or XML

This mining offers manufacturers to connect the information of its own Web site to track routing and bunch information of person into site maps. As a result, person can access the chosen data with facts through keyword association and content mining. Hyperlink hierarchy determines the path associated information contained by the sites connected as well as linked through search engines. Therefore, gathering of related Web pages is done to find the association between these pages

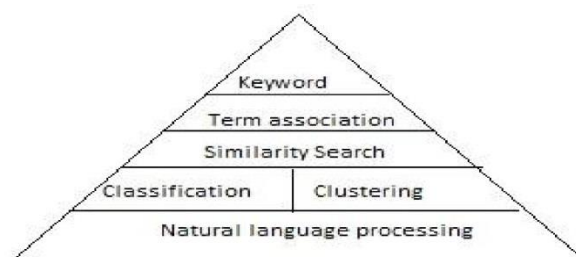


FIGURE 1.2 Text Mining Hierarchies

Method of Web Structure Mining

- Page Rank method
- HITS method

Page Rank

It is the method used by Google. This method calculates the significance of a page and to give significance to pages returned from a conventional search engine by means of keyword probing. The Page Rank [14] significance for a page is measured on the basis of number of pages that point towards it.

Let us suppose we have a page p , we use B_p to be the collection of pages that point towards p and F_p to be the collection of associations out of p . The Page Rank of p is defined as:

$$PR(p) = c \sum_{q \in B_p} PR(q) / N(q) \quad 0 < c < 1,$$

used for normalization

$$|N_q| = |F_q|$$

As soon as a cyclic reference takes place with Page Rank, it gives rise to Rank sink problem

HITS (Hyperlink-induced topic search) algorithm:

A search engine SE, determines a small set, root set (R), of pages P, which contain the known question q. By this, a large collection is made out of it, base set (B), by adding pages associated either to or from R. This is used to get a sub-graph of the Website traversed [14].

$$R = SE(W, q)$$

$$B = R \cup \{\text{pages that link to pages in } R\}$$

$$G_{B,L} = \text{Sub-graph of } W \text{ provoked by } B$$

$$G_{B,L}' = \text{Deleted associations in } G \text{ contained by same site}$$

$$X_p = Y_q; \text{ locating authority weights; } \sum_q \text{ where } q, p \in L'$$

$$Y_p = X_q; \text{ finding hub weights; } \sum_q \text{ where } p, q \in L'$$

$$A = \{p \mid p \text{ has one of the greatest } X_p\}$$

$$H = \{p \mid p \text{ has one of the greatest } Y_p\}$$

1.4 Need of Web Mining

Web mining is vital because for every question comes in mind, we take help of internet and search engines [15]. As the internet and its tradition go on increase and gives a chance to examine web data and mine all useful and important facts from it. The precedent five years have seen the arrival of Web mining as a fast rising area, due to the hard work of the research society as well as different organizations that are involved in it. Web mining decreases the cost and time of the users because it gives the information that is really valuable for the users. With the new exponential expansion of the quantity of content or data on the Internet, it has turn out to be progressively complicated for persons to discover and make use of information in well-organized way. Hence, it becomes difficult for content contributor to categorize and index documents [16]. Conventional web search engines gave lot of outcome for a search, which is lot of time consuming for person to surf/browse and look for concerned data. Online libraries along with other bulky document (e.g. news story archives, customer support databases, press release archives, product specification databases, etc.) are increasing so fast that it is boring and expensive to classify every document physically/manually [18]. So to tackle these problems and situations, scientist are finding computerized technique for functioning with online documents so that they can be more simply analyzed, surfed, planned, and indexed.

1.5 Techniques and Algorithms used in web mining

There are several techniques that are used in web mining. These different tactics and approaches are well considered and applied in various applications and system by research hard work made by the skill of Information Science, Expert System, Natural Language Processing, Database, Human Computer Interaction even Social Science. These procedures and methods [13] are built from the views of a variety of disciplines. Therefore, they are extensively used and functional in the above mentioned areas simultaneously.

1.5.1 Association Rule Mining

Association rule Mining is a method for finding fascinating and main associations among variables in huge databases [17]. Its basic job is to recognize strong rules exposed in databases by means of measuring interestingness along with similarity. Such information can be used for decisions making process i.e. about advertising behavior such as, e.g., pricing or placements.

The reason of discovering association rules is to examine the co-existence connection between set of items, which is then used to build suitable reference. This research has produced a enormous deal of attention through the recent time in data mining because it is the foundation of many applications, such as consumer performance study, stock market analysis, and DNA series investigation.

For instance, a rule $\text{apple} \Rightarrow \text{strawberry}$ (90%) shows that nine out of ten consumers who buy apples also buy strawberry. These set of rules can be helpful for stock market prediction, DNA series investigation etc. [18].

1.5.1.1 Sequence mining

Sequence mining is study of discovering statistically significant, interesting and hidden structures among data set, where the values are distributed in a series. It is often implicit that the values are distinct and therefore series mining is very much connected to sequence mining, but it is usually considered a different activity. Sequence mining is a subset of structured data mining. There are lots of key conventional computational problem given in this domain. These comprise of building proficient databases and catalogs for information, extracting the frequently happening patterns, matching sequences for resemblance and improving omitted sequence members. Generally, sequence extracting problems can be categorized as string mining (deals with a limited alphabet for items) which is normally dependent on string processing algorithms along with item set mining (discovering regularities between frequently co-occurring items) which classically depends on association rule learning.

1.5.2 Supervised learning

Supervised learning is the machine learning task of inferring a function from labeled training data. It is used in almost every area, including text and Web domains [16]. A data set used in the learning task consists of a set of data records, which are described by a set of attributes $K = \{K_1, K_2, K_i | K_i\}$, where $|K|$ denotes the number of attributes or the size of the set K . The data set also has a special target attribute P , which is called the class attribute. In our following discussions, we consider P separately from attributes in A due to its special status, i.e., we assume that P is not in A . The class attribute P has a set of distinct values, i.e., $P = \{p_1, p_2, p_i, |P|\}$, where $|P|$ is the number of classes and $|P|$. A class value is also called a class label. A data set for learning is

simply a relational table of database. Each record in database describes a part of “past experience”. In the machine learning and data mining literature, a data record of database is called an example, an instance.. A data set contains a set of examples or instances. Given a data set X, the objective of learning is to produce a classification/ prediction function to relate values of attributes in K and classes in P [16]. The function can be used to predict the class values/labels of the future data. The function is also called a classification model, a predictive model or simply a classifier.

In order to find the solution of a given problem of supervised learning, one has to perform the following steps:

1. The first task is to determine the type of training examples. For instance, this can be a single handwritten character or a complete line of handwriting.
2. Secondly, gather a training set. The training set needs to be representative of the real-world use of the function.
3. Determine the input feature representation of the learned function. The effectiveness of the learned function mainly depends on how the input object is represented.
4. Determine the structure of the learned function and corresponding learning algorithm.
5. After, completing the design. Run the learning algorithm on the collected training set.
6. Estimate the correctness of the learned function. After parameter alteration and learning, the efficiencies and effectiveness of the resulting function should be measured on a test set that is separated from the training set.

1.5.2.1 Decision Tree Induction

Decision tree learning is utmost extensively technique used for classification [16]. Its classification correctness is competitive with other learning techniques, and it is very effective. The learned classification model is denoted as a tree called a decision tree. Decision tree learning, mostly used in statistics, text mining and machine learning. It uses a decision tree as a predictive model which maps interpretations of an item to arrange the item's objective value. More expressive names for such tree models are classification trees. In these tree arrangements, leaves represent class labels and branches represent conjunctions of features that lead to those class labels.

In decision tree exploration, a decision tree can visually and explicitly represent decisions as well as decision making. In data mining, a decision tree defines data but not decisions; rather than resulting classification of an input for decision making.

1.5.3 Semi-supervised learning

Semi-supervised learning [20] is a machine learning technique that makes use of both tagged and untagged data for training - generally a small amount of labeled data with a large amount of unlabeled data. Semi-supervised learning lies between unsupervised learning (without any labeled training data) and supervised learning (with totally tagged training data).

Bin Hu, Jingzhi Yan, Xiaowei Li, [19] propose to apply the semi-supervised learning technique to expect each individual's information need based on the personalized model. The semi-supervised learning method uses the global information from the labeled and unlabelled words to make predictions. The reality here is that the labeled words from pages marked by each user are very limited while we have large number of data unlabelled. The semi-supervised learning method is especially efficient under this kind of situation. It combines the information from unlabelled words in learning while the supervised learning method only used labeled words in training. As the supervised learning method uses only very limited training data, it won't be very efficient to explore the information from large amount of unlabelled data. Therefore, they construct profile (annotated data) for each user and apply the semi-supervised method to guess its current information need based on its own behaviors.

1.5.3.1 Graph-based methods

Graph-based techniques use a graph depiction to analyze the data, with a node for each and every labeled and unlabeled example. The graph may be created and viewed using field knowledge or resemblance of examples; common method is to attach each data point to its k nearest neighbors

or within some range. The weight w_{ij} of an edge between x and x_{ij} is then set to $e^{-\frac{\|x_i - x_j\|^2}{\epsilon}}$

1.5.4 Unsupervised learning

Unsupervised learning is machine learning method which refers to the problem of finding hidden structure in unlabeled data. Since the instances given to the learner are unlabeled, there is no error or reward signal to evaluate a potential solution. This differentiates unsupervised learning from supervised learning and reinforcement learning.

Sentiment classifications are indicated by the opinion words and phrases [21]. It would be quite natural to use unsupervised learning on such words and phrases. Fixed syntactic phrases are used for classification to express opinions. Density estimation problem is solved using unsupervised learning technique. However, unsupervised learning also comprises of many other methods that seek to précis and explain key features of the data. Its task is to find some grouping of the data. While performing cluster analysis (K-means), the machine is told in advance how many clusters it should form -- a potentially very tedious and arbitrary decision to make.

One of the mostly used methods is clustering for unsupervised learning. Unsupervised classification of patterns (observations, data item, and feature vectors) into clusters (groups) is known as clustering. Clustering in data mining is very valuable and important to find out dispersal patterns in the data. Clustering algorithms uses a distance metric-based similarity measure in order to partition the database such that data points in the same partition are more similar than points in different partitions. Jiayun , Vlado and Gao [22] discuss a system that uses the cluster labels as Web page content indicators after merging web page clustering into log file association mining. The rules can be used further in different applications and systems, including Web user profiling and Web site construction. Their Evaluation shows that character n-gram based clustering implements better than word-based clustering in terms of an internal quality measure. But, word-based cluster profiles are easier to summarize manually.

2.1 Opinion mining

Opinion mining is also recognized as sentimental analysis. It involves building a method to collect and analyze views about the product that are posted and reported in blog, comments, and reviews. Automated opinion mining sometimes use machine learning techniques, a part of artificial intelligence [21].

The exponential growth and development of social media has created many opportunities for people to publicly give their opinions, but has created serious drawbacks/delays in development when it comes to analyze these opinions. At the same moment, there is need to understand the information of people because of the qualities of social media (where attention is very rarely and vastly distributed) as some matters may unpredictably become important very fast. Not only this, manufacturers and people does not have an effective and precise way to analyze sense of this group conversation and interact meaningfully with thousands of other people [27].

Due to which, the public argument in social media is increasing on every topic. Many skilled people think that social media as a missed opportunity for good policy debate. Hence, we face the problem of information overload and can become a issue; therefore it gives a chance for analyzing many voices and identifies issues as they arise.

In today's world, textual information is generally divided into two important parts, facts and opinion. Facts are unbiased expression about entity. Opining is usually subjective expression about their entity. Opinion are very important, as whenever we want to make decision, we gather the opinions from other people. Opinions are not only important for individuals but also important for the organization. Opining mining is a technique used for automatic extracting a knowledge from others opining about a particular problem or topic. It is a part of natural language processing from which we know the feeling of the public about that product [26].

In general, Sentiment study is to find the attitude of a writer/speaker towards a specific subject for determining the polarity of a document [21]. This hostile behavior may be his/her conclusion or affective state, assessment.

2.2 Need of Opinion mining

In recent days we have seen an explosion of data availability, due to the increase in electronic action performed (such as using social networks online) and the tremendous increase of Information technology. The mining opinion is not only associated with the topic of document but also it expresses the opinion of the customer [24]. Due to the limitation of human thinking, combined with the existing simple interfaces available for surfing, discussion and comments, often leads to contradictory statements. To face this challenge [21, 25], opinion mining is dissimilar from data mining as well as text mining as it deals with descriptive statement. The exponential growth of user-generated data enlarges the application scope of public opinion mining tools, which are spreading fast and becoming available to the majority of people, for example:

1. For the people in field of marketing, it can facilitate you in deciding the success of a new product launch or ad campaign and determining which types of a particular product or service are famous and also identifying which particular features are liked or disliked of the product or the service.
2. A sample criticism might be very favorable towards Nikon digital camera, but it can be negative regarding its weight. To be able to identify this kind of information from the sample review in a methodical way gives the manufacturer clear picture of the public opinion regarding the product or service rather than doing surveys or focus groups because the sample review is produced by the purchaser.

2.3 Opinion Mining Applications

Sentiment analysis and Opinion mining cover many applications [21].

1. Argument mapping software is helpful in organizing a rational way from these policy statements, by explicit mapping the rational connections among them. Online Deliberation research field includes tools like Compendium, Cohere, and Debate graph that were made to give a rational structure to a number of statements and to connect arguments with the proof to back it up.
2. Voting Advise Applications assist people in deciding which political party (or other voters) has a closer position. For example, SmartVote.ch asks the elector to give its

degree of conformity with a number of policy statements, and then compares its situation with the political parties.

3. Automated content analysis facilitates in dealing with great quantity of qualitative data. Today lot of tools is available in market that coalesce statistical algorithm with semantics as well as ontology, as well as machine learning with human being direction. These solutions are able to recognize pertinent comments and assign them as positive or negative class to it (the so-called sentiment).

2.4 Challenges in Opinion Mining

There are a number of challenges that exists in opining mining:

1. Multi-Meaning words: Firstly, a word can be positive in particular situation while it can be thought-out negative in other situation, depending upon situation [23]. Let us give an example: Take a sample word say "long" for illustration. If a purchaser said that battery life of his mobile is long, it is a positive opinion. On the other hand, if some other customer said that the start-up time of his mobile is very elongated, it is a negative opinion. These differences in the meaning of a word denote that an opinion system that is skilled to collect opinions/reviews for one type of product or product feature or service may not be competent to perform well.
2. Grouping synonyms: Secondly, in opinion mining people have various ways to express opinions about a particular object [21]. Most conventional content processing techniques depends on the reality that, with the little difference between two pieces of text, the sense of that text does not change so much. However this is not true in case of opinion mining. In opinion mining, "the sound quality is good" is very much different from the "the sound quality is not good".
3. Co-reference resolution: Many of the reviews have both optimistic as well as pessimistic review which is analyzed one by one at a instance [23]. Wenhao Zhang, Hua , Wei Wan [23], extracts feature and groups explicit features but they have not considered the multi-meaningful words and co-reference resolution. However, the more and more informal the medium (for example: blogs, twitter or Facebook), the more probably the people make dissimilar views into one single sentence. For example: "Although the lead artist did well,

the movie bombed is simple for people to understand it, but it is very complicated for a machine to parse it. Occasionally some other people may find complexity in understanding the thought of somebody else based on a small portion of content because it may lack in text explanation. For instance, "That film was outstanding as his last film" is fully reliant on the individual's opinion about the preceding film.

4. Mapping of implied features: Feature indicators are found by feature mining. The most common type of feature indicators are adjectives and adverbs [21]. Particular features or characteristics of objects are described or modified by means of many adjectives and adverbs. For instance, the adjective *heavy* generally tells about the *weight* of an entity, and therefore, it is supposed to be mapped to the *weight* trait. Their precise implication may rely on the field/situation. For instance, "heavy" in the sentence "*The road traffic is heavy*" does not denote the *weight* of the road-traffic.

2.5 Representation of Opinion

Opinion can be represented [21] as a tuple- (F, O, SO, T, H) where,

- F is the features of the object O.
- O is a target object.
- SO is the sentiment value of the opinion proprietor H on feature F of Object O at time T. SO is positive, negative, more granular rating or neutral.
- T is the time when the opinions are stated.
- H represents an opinion holder.

2.5.1 Objectives of Sentiment mining

We have an opinionated document,

- To find out all quintuples (F,O, SO, T, H)
- Unstructured Text is converted to Structured Data

2.5.2 Types of Opinion

- Direct Opinions: These are the sentiment expressions regarding some objects. Objects can be products, topics, events or persons [21].

E.g., — The mobile camera has the great picture quality.

- Comparisons: Comparisons are the relations used to express differences or similarity about more than one entity. Generally, It is used to express an ordering.

E.g., —mobile y is costlier than mobile x.

Chapter 3

Sentiment Mining

3.1 Perception of Sentiment Mining

Sentiment analysis or Opinion mining is the detailed study of opinions, emotions and sentiments expressed in reviews. It is an area of research in which efforts are made to build an automatic system to find human views or behavior from text. It seeks to determine the view point of the people underlying a text span [21].

Generally, feedback data and reviews can be given on anything; it can be a product, an organization, a service, an event, an individual, or a topic. The word object is to signify the target entity about which comments are to be made. An object consists of many attributes (or properties) and many components (or parts). Each component is divided into sub-components, which further has many attributes and so on.

Example of a sample Review: —I bought a Samsung Corby-Mobile Phone a few weeks ago. It is a nice phone and very convenient to use. The touch screen is really very smooth and very cool. The voice quality is extremely clear too. The battery backup is not good, but can compromise with that for its other features. But, my friend was angry on me because I didn't let him know till I purchased the phone. He thought that it is very expensive, and wanted me to return it.

Now the question arises: what we should conclude from the sample review?

- Firstly, we should notice that there are many opinions in this review.
- Secondly, we should note that the opinions have a particular target or object about which the opinions are given.
- Finally, the sources and the holders of the opinions should also be noticed.

Opinion mining is defined as a study of computational semantics that emphasizes on mining opinion of the people from the web [6]. The growth of the internet technology inspires users to donate and precise themselves via videos, social media sites, blogs, etc. All these posts, reviews and reports deliver a huge amount of relevant information that we are interested in analyzing.

Opinion mining tries to mine all these things from a given review text. Suppose we have a part of text, then opinion-mining systems analyze:

- Opinion is expressed by which part?
- Opinion is written by whom?
- What is the object or target being commented?

Sentiment Mining is about finding the partisanship [6] (subjectivity), polarity (optimistic or pessimistic) and polarity strength (weakly positive, mildly positive, strongly positive, etc.) of a review, that is:

- What is the behavior or attitude of the writer?

3.2 Sentiment classification

Sentiment classification is treated as sentiment analysis text classification. It classifies a document (that is., a product review) on basis of polarity i.e. a optimistic or pessimistic opinion [27]. This process is also known as the document-level sentiment classification because it proceeds the whole document as the basic information unit.

Two subtopic that have been extensively studied under sentiment classification [21] are

- 1) Classification of opinioned document and expressing optimistic or pessimistic expression: This aims to find out general opinion of the author
- 2) Categorizing a review/sentence or phrase of the sentence as subjective or objective.

3.2.1 Sentence Level Sentiment Classification

The process of assigning a sentence as opinionated or not opinionated is called subjectivity classification. The resulting opinionated reviews are classified as positive or negative opinions, which is called the sentence level sentiment classification [28]

Sentence-level sentiment analysis has two basic tasks:

Subjectivity classification: To find out whether sentence is Subjective or objective.

Objective: e.g., I bought a Nokia Phone two weeks ago.

Subjective: e.g., it is very nice phone.

Sentiment classification: It means finding the semantic orientation of subjective sentences or phrases, and classify as positive or negative.

3.2.2 Document level sentiment classification

Let D_i be a given a set of review documents, the basic is to find whether each document shows positive and negative expression [21].

- Supervised learning: Classification is done on basis of predefined data. Data is labeled with number of pre-defined classes. It is like that a person gives the guidance (supervision).
- Unsupervised learning (clustering) Class labels of the data are not known. For a given set of data, the objective is to determine the existence of various classes in the data.

3.2.3 Feature level sentiment classification

As we have seen that classifying a review at the document level or at the sentence level is useful in numerous cases, but they do not provide the necessary and significant detail needed for some other applications. A positive review on a particular entity does not denote that the writer or reviewer has positive opinions on all aspects or features of the entity. Similarly, a negative review does not mean that the writer or reviewer dislikes the whole thing. In a review of manufactured goods, the writer writes both positive and negative features of the entity, but the general response/feeling i.e. of the whole document may be positive or negative. Document-level and sentence-level classification does not make available such information that is about the features. To get hold of such details, we require to go to the object feature level. At the feature level, the mining job is to locate every quintuple $(o_j, f_{jk}, oo_{ijkl}, h_i, t_l)$ and identify all the synonyms (W_{jk}) and feature indicators I_{jk} of feature f_{jk} . In this section, we mainly focus on two key mining tasks [21]:

1. Recognize the sentiment i.e. thoughts of the opinion proprietor based on the features of a particular object. Because a person may like some features and some other person may not like that.
2. It gives enhanced and in depth analysis of the product than the sentence level and document level sentiment categorization.

3.3 WordNet Similarity Measures

WordNet is a great lexical database of English verbs, nouns; adjectives as well as adverbs are grouped into sets of cognitive synonyms (synsets), each one giving a unique concept [33]. Synsets are interconnected by means of conceptual-semantic and lexical relations. Word net::Similarity is a software package which is freely obtainable. It compares two concepts and gives the measure of similarity and relatedness connecting them. It provides six measures of similarity and three measures of relatedness. The measures are based on lexical database Word Net. It takes the two concepts as input and returns the degree to which they are interrelated

Information Content (IC) is a measure of specificity for concept information [32]. The Content is calculated based on frequency counts of concepts as found in a corpus of text. The number of counts is incremented in WordNet each time that concept is observed. Out of the six measures of similarity measures, three are based on the information content of the least common subsumer (LCS) of concepts A and B. These measures are res (Resnik, 1995), lin[35] and jcn[34]. The lin and jcn measures augment the information content of the LCS with the summation of the information content of concepts A and B themselves. The lin measure scales the information content of the LCS by this sum, while jcn takes the variation of this sum and the information content of the LCS. The Resnik measure simply uses the Information Content of the LCS as similarity value whereas the Lin and Jiang and Conrath measures try to improve the Resnik measures in two different ways.

$$\text{Res}(C1, C2) = \text{IC}(\text{LCS}(C1, C2))$$

$$\text{Lin}(C1, C2) = 2 * \text{res}(C1, C2) / \text{IC}(C1) + \text{IC}(C2)$$

$$\text{Jcn}(C1, C2) = 1 / \text{IC}(C1) + \text{IC}(C2) - 2 * \text{res}(C1, C2)$$

The additional three measures are based on path lengths connecting the two concepts A and B. These measures are lch (Leacock and Chodorow, 1998), wup (Wu and Palmer, 1994) and path.

Opinion mining is basically concerned with identifying opinion words from reviews i.e. nice, good, bad, beautiful, and great. Countless researchers have worked on extracting such terms and recognizing their polarities.

4. 1 Opinion mining of Product Reviews by means of Association Rules

Numerous citizens make inquiries regarding other people's opinion before buying a product and then refer to a suggested product. Nowadays, the outcome of tremendous growth of the Web makes it straightforward and easy to find other people's opinion information. These various types of opinion data are not only helpful to customers, but also valuable to manufacturers. As a result, opinion mining study is to explore opinion data on the internet and has become admired topic these days. Ryu et al [50] have used POS tagging for extracting features, then discovered association rules and provided information using PMI-IR algorithm. The sentiment analysis plays a critical role, and it is being comprehensively studied and discussed since 1990s [36, 40]. There are primarily two essential approaches to carry out sentiment analysis study, first is based on semantic analysis [38, 40, 41 42], and the second one is based on machine learning [37, 43]. The document-level sentiment analysis commonly uses machine learning technique.

4.1.1 POS Tagging

Part-of-speech (POS) labeling is the method of allocating a part-of speech like pronoun, noun, adjective, adverb and additional lexical class indicator to each term in a sentence. The input to a labeling algorithm is a sequence of words of a natural language sentence along with a limited list of Part-of-speech labels. The output is a distinct POS label for each word. Stanford Tagger [45] is an application that analyzes sentences for POS tagging. After POS tagging on a sentence, data structure is needed to analyze the sentence. Stanford Parser [46] is an application that stores tagged sentence in data structure form of phrase-structure tree.

4.1.2 Association Rule Mining

Association rule mining is a very famous and well researched method in data mining for exploring remarkable association among variables in large databases. Aggrawal et al was the initial one to introduce Association rule mining [48]. Association rule mining is a way to discover out association rules that assure the predefined minimum confidence and support from a specified database.

Let $K = K_1, K_2, \dots, K_n$ be a set of n different attributes, P be transaction that has a collection of items, such that $P \subseteq K$, Z is a database that consists of transaction T_s . An association rule is an inference of the form $A \subseteq B$, where $A, B \rightarrow K$ are set of items called item sets, and $A \cap B = \phi$. Association rules are measured by two important terms Support and confidence.

Support of a rule is ratio of the percentage of tuples that comprise $A \cup B$ to the total number of tuples in the database.

Confidence of rule is ratio of the percentage of the number of transactions that contain $A \cup B$ to the total number of tuples that comprise A .

Support finds how often a rule is appropriate to given set of data, and confidence finds of strength of association rules.

Kim et al [50], had done POS labeling on every review sentence and then extracted feature along with opinion terms in the of transaction data form. Then they discovered association rules which were required from the transaction records, and present statistics that summarizes merits and demerits using PMI-IR algorithm.

4.1.3 Semantic Orientation from Association

The semantic orientation (SO) of opinion is used to categorize reviews as optimistic or pessimistic. The pointwise Mutual Information (PMI) among two terms i.e. term1 and term2, is given as follows [49].

$$\text{PMI}(\text{word1}, \text{word2}) = \log_2 \left[\frac{p(\text{word1} \& \text{word2})}{p(\text{word1}) p(\text{word2})} \right] \quad (1)$$

The semantic orientation of a given term is determined from the difference between potency of its link with a set of positive terms and strength of its link with a set of negative terms. It is determine as follows

$$SO (PHARSE) = \log_2 \left[\frac{\text{hits (phrase NEAR "excellent")} * \text{hits ("poor")}}{\text{hits (phrase NEAR "poor")} * \text{hits (" excellent")}} \right] \quad (2)$$

4.2 Product weakness from Feature based sentiment analysis

Determining the flaw of products from the response of the customers can assist manufacturers to enhance their quality of product and competitive potency. In recent days, many people convey their views about the products on web, which can be easily gathered. Also, a company A can collect the reviews of competitor’s product to compare the performance of their product with other one. On the other hand, it’s not possible for manufacturers to interpret each feedback of a customer to analyze the weak point of their products and services. Consequently, determining product weak point from online feedback has become a significant work. Zhang et al [23] initiated such an expert system, Weakness Finder, which can assist producers to discover out their product weak point by using features based sentiment exploration. A feature is a component or attribute of a product, such as cost, smell are the features of the body shower product. Wenhao Zhang, Hua, Wei Wan [23], uses morpheme based method to extracts feature and groups explicit features by using HowNet dictionary similarity measure, then integrated the implied features with collocation assortment technique for each feature. Then make use of sentence based sentiment analysis technique to determine the semantic orientation of each feature in sentences. But, they have not considered the multi-meaningful words and co-reference resolution.

4.2.1 Grouping features

Integrating the features into various aspects is to combine the words that portray the same aspect together [35]. We identify that some expressions like “cost” and “cost level” can explicitly describe the aspect “cost”, while certain terminology like “cost-saving”, “low-priced” and “reasonable” can suggest the feature impliedly. Wan et al [23] had introduced an algorithm to combine two types of features correspondingly (a) Explicit features (b) Implied features.

Naturally, the explicit features or aspects in the similar group are generally the substitutes, antonyms, or the similar concept words [23]. The approach to discover synonyms and antonyms is to glance up synonyms and antonyms dictionary, it can present specific synonyms and antonyms words, for example, the synonyms of “cleanness” are “hygienic” and “cleanse” and the antonym is “filthy”. Nevertheless, results are all the time restricted due to the expansion of vocabulary and some terminology may not be synonyms according to dissimilar domains. In our system we have used wordnet similarity, for grouping explicit features. We have set a threshold value, for combining the features, which that if similarity is greater than a particular threshold value then we, will combine that feature, otherwise not.

4.3 Feature and Opinion Mining

In current year, due to presence of blogs, discussion groups and numerous forums single users are joining more enthusiastically and are producing vast amount of new data – termed as *user-generated contents* [6]. These new online contents include purchaser feedback and blogs that express views on products and services – which are jointly referred to as purchaser feedback data on the internet. Consumer feedback data on the internet, impacts on decision of other consumer, these reviews have become an significant basis of information for industries to take vital decision on developing effective marketing and product development strategies. Muhammad Abulaish, Jahiruddin, MN Doja, Tanvir Ahmad [51] proposed an opinion mining system to recognize product features and opinions from review documents. The features as well as opinions are mined using semantic and linguistic exploration of text documents. In a bootstrapping approach is suggested [52], which uses a small collection of particular seed opinion words to find out their synonyms and antonyms in WordNet. The semantic orientation of opinion sentences is recognized using polarity scores of the opinion terms through Senti-WordNet to produce a feature-based summary of review documents. The system is also united with a visualization module to present feature- based summary of review documents in a significant way. But they have not refined the rule-set to increase the correctness of the system.

4.4 Mining and Summarizing Customer Reviews

B liu and Minqing Hu [52] have proposed a method for feature-opinion summarization. This summarization task differs from outdated text summarization as they only define the aspect of

the product on which the consumers have conveyed their thoughts and whether these views are negatively or positively stated. They do not summarize the reviews by selecting a subcategory or redraft some of the original sentences from the review data set to find out the main points as in the typical text summarization.

4.4.1 The task is performed in three steps:

- (1) Mining product features which are stated by consumers;
- (2) Recognizing opinion sentences in each feedback and determining whether each opinion sentence is negative or positive.
- (3) Results summarization.

4.4.2 Challenges

Conversely, identifying opinion sources and analyzing them on the internet is still a difficult and tedious task because there are a lot of dissimilar sources, and each source may also have a lot of opinionated terminology (text with opinions or sentiments). In numerous cases, thoughts are unseen in long forum posts and blogs. It is problematic for a human reader to find out relevant sources, mine associated sentences with opinions, read them, summarize them and organize them into usable forms.

Thus, robotic opinion detection and summarization systems are required. It is a very stimulating text mining problem or natural language processing. But they have not considered the pronoun resolution i.e. co-reference resolution and strength of opinions.

4.5 Examining and comparing opinion in Web

The web has become the important source of information to collecting a customer feedback. There are large no of web site containing such opinion, e.g. customer review of product and blogs. Bing liu, Hu M, and Cheng [53] all are focusing to determine online customer review on product. They have made two contributions. First they propose novel frame work for analysis the customer review of competing product. A prototype system is called opinion observer is also applied.

The system is such that with single look of its visualization, the user is able to understand the strength and weakness of the product in the mind of the consumer about several product features. This visualization is beneficial for both side that is consumers and the makers. The customers can see all the product feature and consumer opinion about the product that he/she wanted to buy which will help him/her in buying the product. For product manufactures, the comparison from visualization helps it to get marketing effectively easily. Second method established on language pattern mining purposed to extract product feature pros and cons of specific kind of feedback [53].

4.5.1 Two challenging task required to be implemented

1. Recognizing the product feature on which customer expresses their positive and negative opinion about the product.
2. For each feature identify the opinion whether it is positive or negative, positive means customer is satisfied and negative opinion mean complain about product feature

There are three review format of web

1. Pros and cons: this review is asking to describe pros and cons separately.
2. Pros, cons and detailed review: this review asks to describe the pros and cons separately and in detail.
3. Free review: this review write freely or we can say that no separate pros and cons,

4.5.2 Opinion Observer works in two stages

The consumer review can be mined and examined in following two step

Step 1:-

1. In this case it connects and automatically downloads all the review on the page.
2. In this stage, all the new reviews (which were not examined before) of every product are examined. The two tasks perform from each review is to recognize the product attributes and opinion orientations. It can be done mechanically or semi mechanically.

Step 2:-

In this step, it is based on the study outcome; a variety of consumers can imagine and judge against opinions of various products using a consumer interface. The consumer just picks the products that he/she wishes to judge against and the systems then recovers the examined outcome of these products and shows them in the interface.

4.6 Clustering Product Features for Opinion Mining

In sentiment analysis of product reports, most significant trouble is to make a synopsis of opinions based on product features/characteristics. though, for the exact attribute, people can convey it with several various types of words or phrases. for a useful synopsis, these words and group of words, which are area synonyms, need to be grouped under the same feature set. Even though several techniques have been made to mine product features from feedbacks, restricted work has been done on assembling or grouping of synonym features. Zhongwu Zhai, Bing Liu, Xu, Jia [35] modeled a semi-supervised learning problem. Lexical characteristics of the problem are abused to automatically classify some labeled examples. In sentiment study of product response, one significant trouble is to produce a synopsis of opinions based on product features/characteristics (also called aspects). but, for the same feature, people can convey it with several dissimilar types of words or groups of words. For a useful summary, these words and phrases, which are area synonyms, need to be grouped under the same feature set. Even though several techniques have been made to mine product features from feedbacks, restricted work has been done on assembling or grouping of synonym features.

4.6.1 Product Feature Grouping

Grouping feature expressions, which are domain synonyms, is critical for effective opinion summary [31]. Since there are typically hundreds of feature expressions that can be discovered from text for an opinion mining application, it's very time-consuming and tedious for human users to group them into feature categories. Some automated assistance is needed. Unsupervised learning or clustering is the natural technique for solving the problem. The similarity measures used in clustering are usually based on some form of distributional similarity [31, 41].

5.1 Feature Based Sentiment Mining

When the author writes a opinioned text, it can have both optimistic and pessimistic aspects of the entity, but the general attitude on the entity may be positive or negative. Document-level and sentence-level classification does not deliver any information about the features. To extract such specifics, we need to go for the entity feature level. This method will identify the sentiment of the opinion holder based on the features of a particular object. Because a person may like some features and some other person may not like that.

It gives better and in depth analysis of the product than the sentence level and document level sentiment classification. But, at the feature level, the mining task is to find out every quintuple $(o_j, f_{jk}, oo_{ijkl}, h_i, t_i)$ and recognize all the synonyms (W_{jk}) and aspect attributes I_{jk} of feature f_{jk} . In this part, we mainly focus on two key mining tasks:

1. Recognize the sentiment of the opinion holder based on the features of a particular object. Because a person may like some features and some other person may not like that.
2. It gives better and in depth analysis of the product than the sentence level and document level sentiment classification.

5.2 Proposed Architecture

Let $X = \{X_1, X_2, \dots, X_n\}$ be a set of products which are prepared by various industrialists like 'Nokia', 'micromax' and 'karbon', they are all cell phone but made by various enterprises. For each product X_i , there exists a set of reviews $E_i = \{E_1, E_2, \dots, E_m\}$. And for each review E_j , it may consist of some opinionated sentences about the product's features. Therefore, let $F = \{F_1, F_2, \dots, F_N\}$ be a set of aspects of product, such as "connectivity", "battery backup" and "memory-card", etc.

Problem Definition: We have given a set of opinion/reviews from various companies' products X. The first phase of task is to identify, group and pronoun recognition of the features words f for each aspect A. Secondly, find out the pair of each aspect and its sentiment $O = (F,S)$.

Dataset: The data set we have used is taken from UCI repository, eopinions.com.

5.2.1 Feature Extraction

Stanford Parser [26] marks parts-of-speech (POS) tags to every word based on the context or usage in the sentence. It is basically used to identify the nouns i.e. the Product features, adjectives i.e. opinions, adverbs that is used to express degree of expressiveness of opinions.

5.2.2 Procedure

The algorithm works in five steps:

Input to the algorithm is written customer review.

Output is the Classification i.e. positive or negative.

Steps

1. Use part-of-speech tagger to identify features of the product.
2. Grouping /Clustering similar features using Word Net similarity
3. Using syntactic parse tree and then finding pronoun being referred i.e.co-reference resolution
4. Creating Dependency relations for feature-opinion pair extraction.
5. Estimating the semantic orientation of feature-opinion word.
6. Classifying the given review to a class (positive, negative, neutral)

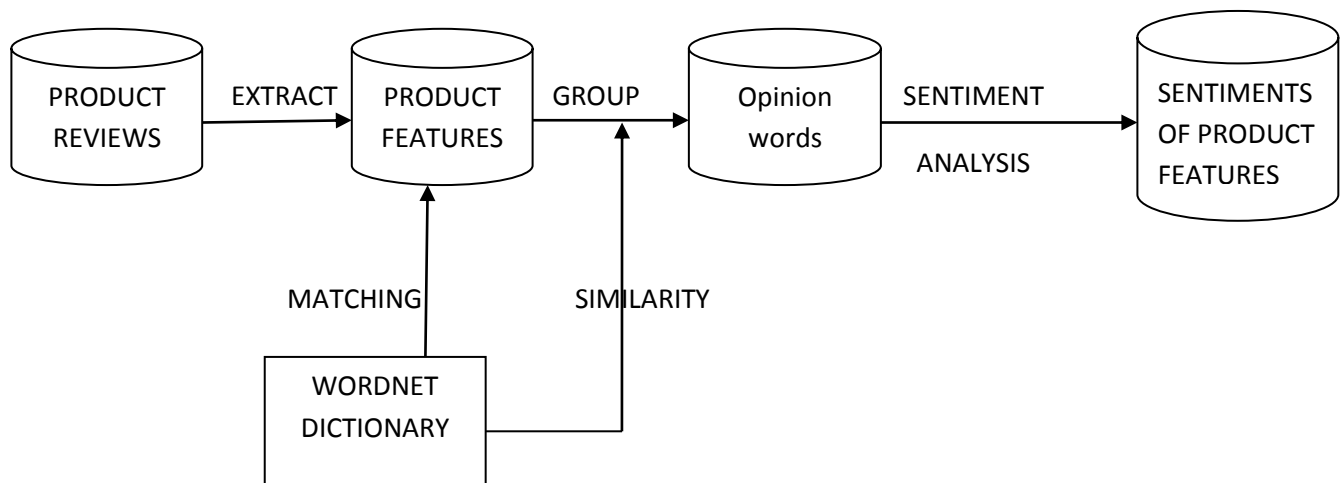


Figure 5.1 Proposed Architecture of our system

All these steps are discussed in detail.

Step 1: Using part-of-speech tagger to identify features of the product.

In this step, reviews that contain noun are extracted. The reason for doing this is that noun is the good indicators of subjectivity. Product features are mainly nouns or noun phrases in data set. Thus the part-of-speech tagging is very important step. We have used the NLP processor(Stanford tagger) to parse the entire review, to fragment text into sentences and to produce the part-of-speech label for each expression. The algorithm [27] used for extracting features is as follows:

Algorithm. Pseudo-Code for extracting product feature candidates

//**Input:** S – Set of tagged sentences; $s = s_1, s_2, \dots, s_m$

P – Set of noun phrase patterns

GI – Set of word in GI dictionary

//**Output:** PS – Set of product feature candidates

PS = \emptyset

For each tagged sentence s_n in S

PC = \emptyset

For $i=1$ to end of sentence s_n

If $i < \text{Length}(s_n) - 2$ Then $x = 3$

Else If $i = \text{Length}(s_n) - 2$ Then $x = 2$

Else If $i = \text{Length}(s_n) - 1$ Then $x = 1$

Else $x = 0$

End

End

End

For $j = x$ to 0

GT = T_i to T_{i+j} /* POS Tag of word_i to word_{i+j} of s_n */

GW = word_i to word_{i+j}

If GT in P and GW is not in GI then

$i = i+j$

PC = PC + GW

Break

End

End

End

PS = PS + PC

End

After that, we have displayed the features as shown in figure 2.

Table 1. Patterns of POS tags for extracting two-word phrases

	First word	Second word	Third word (Not Extracted)
1.	JJ	NN or NNS	anything
2.	RB, RBR, or RBS	JJ	not NN nor NNS
3.	JJ	JJ	not NN nor NNS
4.	NN or NNS	JJ	not NN nor NNS
5.	RB, RBR, or RBS	VB, VBD, VBN, or VBG	anything

where,

JJ - Adjective

NN - Noun, singular

NNS- Noun, plural

RB - Adverb

RBR- Adverb, comparative

RBS - Adverb, superlative

VB- Verb, base form

VBD- Verb, past tense

VBG - Verb, present participle

VBN - Verb, past participle

Part-of-Speech Tagging (POS)

Product features are mainly nouns or noun phrases in opinionated sentences. The process also identifies simple noun and verb groups. The following shows a sentence with POS tags.

```
<S> <NG><W C='PRP' L='SS' T='w' S='Y'> I </W> </NG> <VG>  
<W C='VBP'> am </W><W C='RB'> absolutely </W></VG> <W  
C='IN'> in </W> <NG> <W C='NN'> awe </W> </NG> <W  
C='IN'> of </W> <NG> <W C='DT'> this </W> <W C='NN'>  
camera </W></NG><W C='.'> . </W></S>
```

NLP processor generates XML output. For instance, <W C='_NN'> indicates a noun and

<NG> specifies a noun group/noun phrase. Each sentence is kept in the review database beside the POS tag information of each term in the sentence. A transaction file is then produced for the generation of regular features in the next step. In this file, each line contains words from one sentence, which includes only the recognized nouns and noun phrases of the sentence. Other constituents of the sentence are improbable to be product features. Some pre-processing of expressions is also implemented, which includes elimination of stop words and lessening.

- Stanford POS Tagger is used for tagging the text
 - <http://nlp.stanford.edu/software/tagger.shtml>
 - Class Maxent Tagger is used to for tagging the text file.
 - The output is in the form of WORD/TAG e.g. Nice/JJ – means –nicell is Adjective.
 - Once the tagged text is available, find out the two consecutive words which have the tags mentioned in the above table.
- Below diagram describes the POS tagging process.

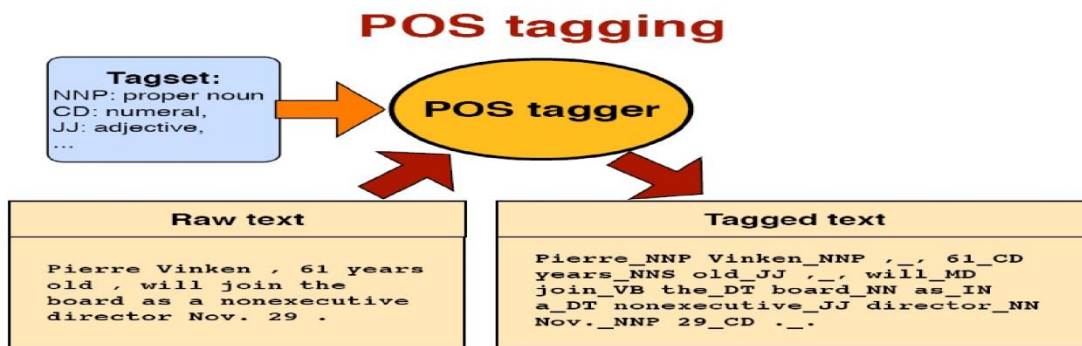


Figure 5.2 POS tagging

Step 2: Grouping /clustering similar features using Word Net similarity

ALGORITHM:

```

For each feature f (k) stored in feature_list
For each feature f (j) [where j =k+1] in feature_list
If the similarity between f (k) and f (j) > threshold then
Group f(k) and f(j)
End if
end for
  
```

end for

Step3: Using syntactic parse tree and then comparing the two parse for pronoun recognition (co-reference resolution)

ALGORITHM:

procedure find(tree,node):

 label node as traversed

 if node = tagged_Noun

 then co_refer = node

 if node = tagged_PRP

 then co_refer = co_feature // assign latest noun node found to pronoun.

 else

 for total edges e in tree.adjacentEdges (node) do

 if edge e is untraversed then

 w ← Tree.adjacentnode(node,e)

 if node w is un-traversed then

 mark e as a traversed edge

 recursively call find(tree,w)

 else

 mark e as a back edge

After finding the nodes we will assign the nouns found to the pronoun being used for them. This process goes on until we find the next sentence having noun and we will repeat the same process. We have used Stanford parser for generating the full syntactic parse tree of the given sentence. After generating the parse tree, we have compared the leftmost branches of parse tree to determine Pronoun in a sentence with its previous adjacent parse tree having Noun. Therefore, we will be able to determine the nouns that are being referred by pronoun.

Step 4: Creating Dependency Relation for feature-opinion pair extraction.

This step is to identify product feature-opinion candidates. For each product feature candidate in every dependency parse tree, we search for the related opinion words. Sentence structures are represented by dependency grammars as a set of dependency relationships. A dependency relationship is an asymmetric

binary relationship between a word called governor or head, and other word called modifier or dependent. The dependency tree consists of dependency of words [27].

Step 5: Estimating the semantic orientation of feature-opinion phrase.

- Calculate the polarity of the extracted phrase using the Point wise Mutual Information (PMI) measure.
- PMI between 2 words, word-1 and word-2 can be defined as:.

$$PMI(\text{word1}, \text{word2}) = \log_2 \left[\frac{p(\text{word-1} \& \text{word-2})}{p(\text{word-1}) p(\text{word-2})} \right] \quad (1)$$

Here,

1. $P(\text{word-1} \& \text{word-2})$ = Probability that both words occurs together.
 2. $P(\text{word-1}) * P(\text{word-2})$ = Probability of co-occurrence of word1 and word 2, If both words are independent.
 3. $\frac{P(\text{word-1} \& \text{word-2})}{P(\text{word-1}) * P(\text{word-2})}$ = Degree of statistical dependence between words.
 4. Log = Gives information of presence of one word when we observe other.
- The polarity of a given phrase is estimated by matching its resemblance to a positive reference word (“excellent”) with its resemblance to a negative reference word (“poor”).
 - More precisely, phrase is allocated a numerical rating or value by taking the common information between the given phrase and the word “excellent” and subtracting the common information between the given phrase and the word “poor”.
 - Not only determining the direction of the phrase’s polarity (positive or negative, based on the sign of the rating), but also this numerical rating or value also indicates the strength of the semantic orientation (based on the value of the number).
 - The Semantic Orientation (SO) of a phrase is calculated as :

$$SO(\text{phrase}) = PMI(\text{phrase}, \text{“excellent”}) - PMI(\text{phrase}, \text{“poor”})$$

When, SO is +ve: phrase is strongly associated with excellent.

SO is -ve: phrase is strongly associated with poor.

- The probabilities are evaluated by issuing queries to a search engine and collecting the number of hits.
- Proximity Search –
 - Search the words such that they are located within ‘n’ words of one another.
 - EXALEAD is search engine which allows Proximity search using “/n” or “NEAR” operator.
 - Query such as “Good Camera /10 Excellent” will find out the occurrence of “Good Camera” with the word “Excellent”
 - Based on the number of this returned by the search query, the PMI can be calculated as –

$$SO (PHARSE) = \log_2 \left[\frac{\text{hits (phrase NEAR “excellent”) * hits (“poor”)}}{\text{hits (phrase NEAR “poor”) * hits (“ excellent”)}} \right] \quad (2)$$

Step 6: Assign the given review to a class

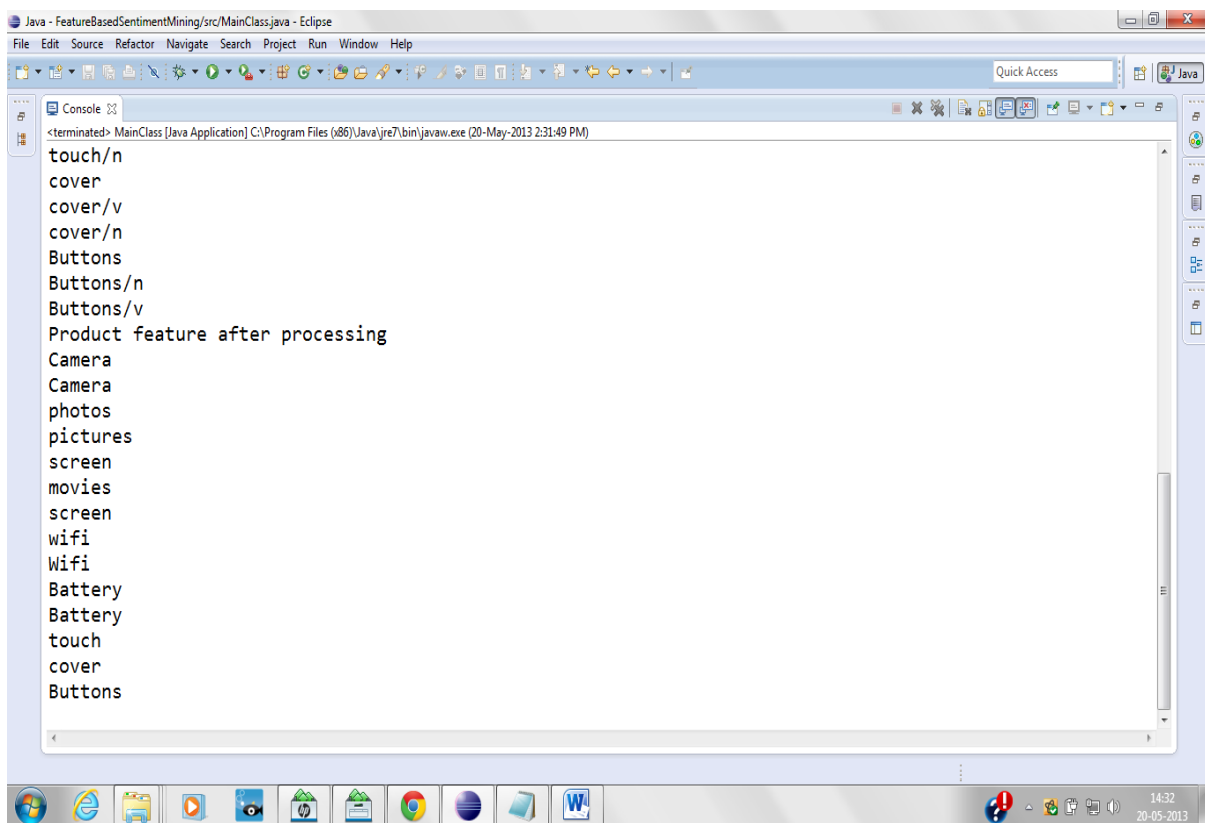
- Estimate the average Semantic Orientation (SO) of the product feature present in the review text.
- Assign them as recommended or not recommended.
- If the average SO is greater than zero then it is Recommended or Positive review.
- If average SO is less than zero then it is not Recommended or Negative review.

6.1 Results

SAMPLE INPUT 1: SAMSUNG GALAXY (PHONE)

The Camera works great. It takes good photo and picture. It has nice screen to watch movies. It is not good. The Battery is not good. It becomes hot. The touch works nice. The cover is not pleasant. The Buttons are not smooth. The Wi-Fi works fast. It gets connected easily.

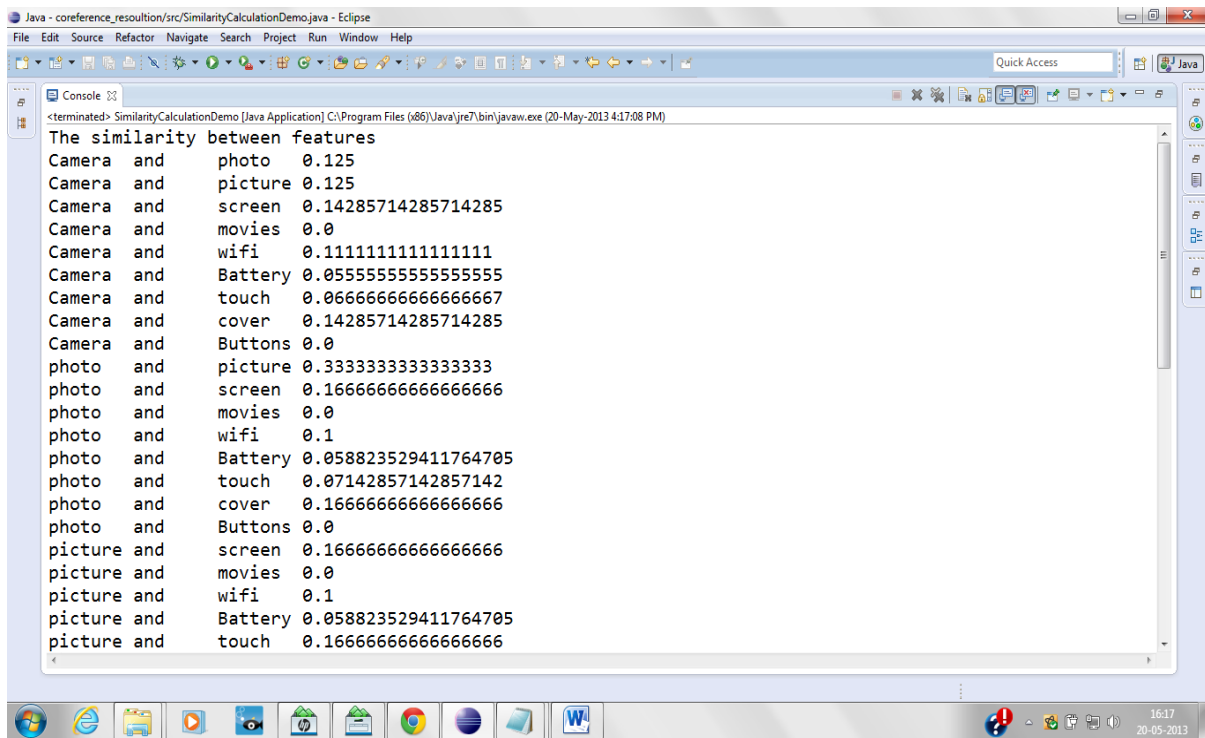
STEP 1: Product Feature Extraction



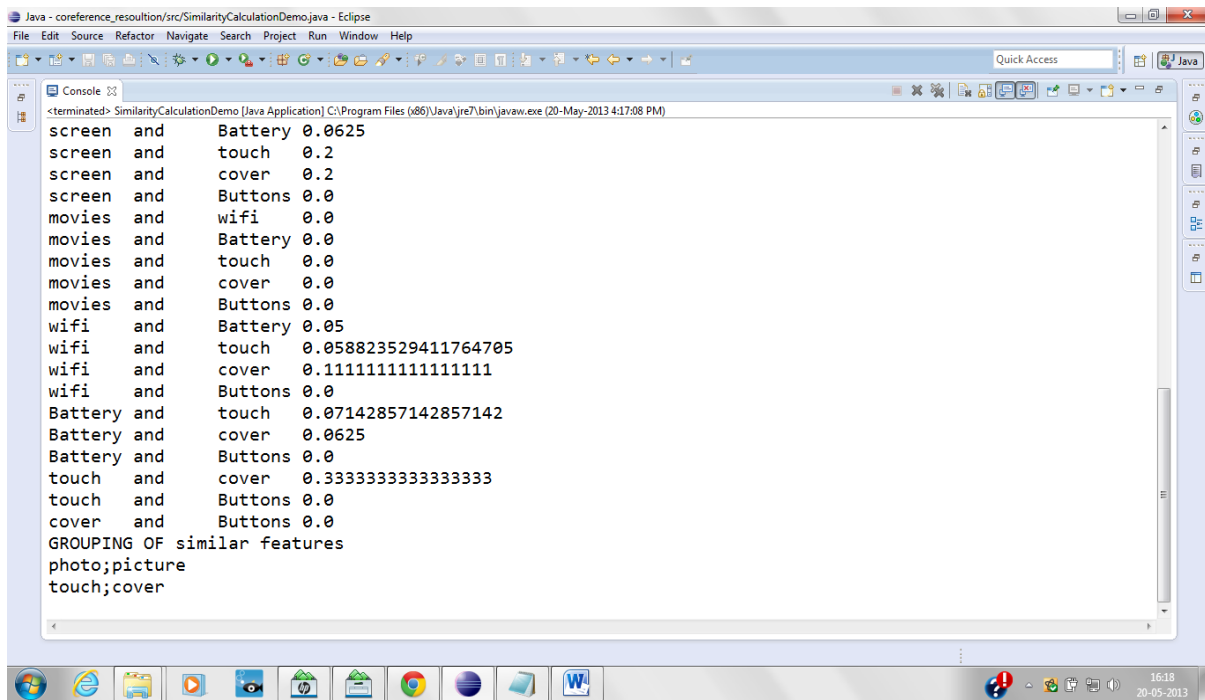
```
Java - FeatureBasedSentimentMining/src/MainClass.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Console
<terminated> MainClass [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 2:31:49 PM)
touch/n
cover
cover/v
cover/n
Buttons
Buttons/n
Buttons/v
Product feature after processing
Camera
Camera
photos
pictures
screen
movies
screen
wifi
Wifi
Battery
Battery
touch
cover
Buttons
```

Figure 6.1 Product Features

STEP 2: Grouping Similar Features using WordNet Dictionary



```
Java - coreference_resolution/src/SimilarityCalculationDemo.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> SimilarityCalculationDemo [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 4:17:08 PM)
The similarity between features
Camera and photo 0.125
Camera and picture 0.125
Camera and screen 0.14285714285714285
Camera and movies 0.0
Camera and wifi 0.11111111111111111
Camera and Battery 0.05555555555555555
Camera and touch 0.06666666666666667
Camera and cover 0.14285714285714285
Camera and Buttons 0.0
photo and picture 0.3333333333333333
photo and screen 0.16666666666666666
photo and movies 0.0
photo and wifi 0.1
photo and Battery 0.058823529411764705
photo and touch 0.07142857142857142
photo and cover 0.16666666666666666
photo and Buttons 0.0
picture and screen 0.16666666666666666
picture and movies 0.0
picture and wifi 0.1
picture and Battery 0.058823529411764705
picture and touch 0.16666666666666666
```



```
Java - coreference_resolution/src/SimilarityCalculationDemo.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> SimilarityCalculationDemo [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 4:17:08 PM)
screen and Battery 0.0625
screen and touch 0.2
screen and cover 0.2
screen and Buttons 0.0
movies and wifi 0.0
movies and Battery 0.0
movies and touch 0.0
movies and cover 0.0
movies and Buttons 0.0
wifi and Battery 0.05
wifi and touch 0.058823529411764705
wifi and cover 0.11111111111111111
wifi and Buttons 0.0
Battery and touch 0.07142857142857142
Battery and cover 0.0625
Battery and Buttons 0.0
touch and cover 0.3333333333333333
touch and Buttons 0.0
cover and Buttons 0.0
GROUPING OF similar features
photo;picture
touch;cover
```

Figure 6.2 Grouping similar features

STEP 3:

Using syntactic parse tree and then comparing the two parse for pronoun recognition i.e. co-reference resolution

```
<terminated> StanfordCoreNlpDemo [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 2:14:51 PM)
Sentence #1 (5 tokens):
The Camera works great.
[Text=The CharacterOffsetBegin=0 CharacterOffsetEnd=3 PartOfSpeech=DT Lemma=the NamedEntityTag=0] [Text=C
(ROOT
  (S
    (NP (DT The) (NN Camera))
    (VP (VBZ works)
      (ADJP (JJ great)))
    (. .)))

det(Camera-2, The-1)
nsubj(works-3, Camera-2)
root(ROOT-0, works-3)
acomp(works-3, great-4)

Sentence #2 (7 tokens):
It takes good photo and pictures.
[Text=It CharacterOffsetBegin=23 CharacterOffsetEnd=25 PartOfSpeech=PRP Lemma=it NamedEntityTag=0] [Text=
(ROOT
  (S
    (NP (PRP It))
    (VP (VBZ takes)
      (NP
```

```
Java - coreference_resolution/src/StanfordCoreNlpDemo.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> StanfordCoreNlpDemo [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 2:14:51 PM)
Sentence #2 (7 tokens):
It takes good photo and pictures.
[Text=It CharacterOffsetBegin=23 CharacterOffsetEnd=25 PartOfSpeech=PRP Lemma=it NamedEntityTag=0] [Text=
(ROOT
(S
(NP (PRP It))
(VP (VBZ takes)
(NP
(NP (JJ good) (NN photo))
(CC and)
(NP (NNS pictures))))
(. .)))

nsubj(takes-2, It-1)
root(ROOT-0, takes-2)
amod(photo-4, good-3)
dobj(takes-2, photo-4)
dobj(takes-2, pictures-6)
conj_and(photo-4, pictures-6)

Sentence #3 (5 tokens):
The Wifi is good.
[Text=The CharacterOffsetBegin=57 CharacterOffsetEnd=60 PartOfSpeech=DT Lemma=the NamedEntityTag=0] [Text=
```

```
Java - coreference_resolution/src/StanfordCoreNlpDemo.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> StanfordCoreNlpDemo [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 2:14:51 PM)
Sentence #2 (7 tokens):
It takes good photo and pictures.
[Text=It CharacterOffsetBegin=23 CharacterOffsetEnd=25 PartOfSpeech=PRP Lemma=it NamedEntityTag=0] [Text=
(ROOT
(S
(NP (PRP It))
(VP (VBZ takes)
(NP
(NP (JJ good) (NN photo))
(CC and)
(NP (NNS pictures))))
(. .)))

nsubj(takes-2, It-1)
root(ROOT-0, takes-2)
amod(photo-4, good-3)
dobj(takes-2, photo-4)
dobj(takes-2, pictures-6)
conj_and(photo-4, pictures-6)

Sentence #3 (5 tokens):
The Wifi is good.
[Text=The CharacterOffsetBegin=57 CharacterOffsetEnd=60 PartOfSpeech=DT Lemma=the NamedEntityTag=0] [Text=
```

```

Java - coreference_resolution/src/StanfordCoreNlpDemo.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Console
<terminated> StanfordCoreNlpDemo [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (21-May-2013 6:44:24 AM)
(ROOT
(S
(NP (PRP It))
(VP (VBZ works)
(ADJP (JJ nice)))
(. .)))

nsubj(works-2, It-1)
root(ROOT-0, works-2)
acomp(works-2, nice-3)

Coreference set:
(2,1,[1,2]) -> (1,2,[1,3]), that is: "It" -> "The Camera"
Coreference set:
(15,1,[1,2]) -> (14,2,[1,3]), that is: "It" -> "The Bluetooth"
Coreference set:
(12,2,[1,3]) -> (3,2,[1,3]), that is: "The wifi" -> "The Wifi"
(13,1,[1,2]) -> (3,2,[1,3]), that is: "It" -> "The Wifi"
Coreference set:
(6,1,[1,2]) -> (5,3,[1,4]), that is: "It" -> "The nice screen"
Coreference set:
(8,1,[1,2]) -> (7,2,[1,3]), that is: "It" -> "The Battery"

```

Figure 6.3 Pronoun reference

STEP 4: Creating Dependency relation for feature-opinion pair extraction

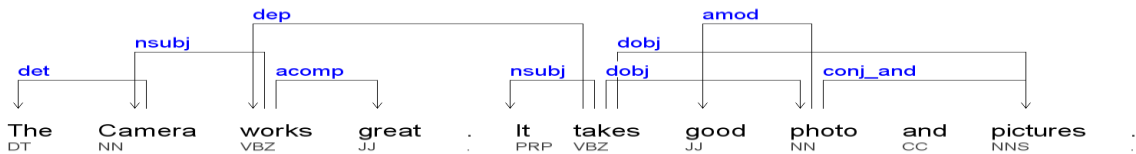


Figure 6.4 Dependency relation

STEP 5: Estimating the semantic orientation of feature-opinion phrase.

(b) Classify the given review to a class (positive, negative, and neutral) by formula [28]

$$SO(\text{PHARSE}) = \log_2 \left[\frac{\text{hits}(\text{phrase NEAR "excellent"}) * \text{hits}(\text{"poor"})}{\text{hits}(\text{phrase NEAR "poor"}) * \text{hits}(\text{"excellent"})} \right]$$

```
<terminated> FeatureAnalysis [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 1:41:37 PM)
POS tagged text: Wifi_NNP connects_VBZ easily_RB ._.
Wifi          connects_VBZ easily_RB

Feature: WIFI
Opinion: connects easily
Result : 3.308746814727783      (+) POSITIVE Feature
*****

POS tagged text: Battery_NNP not_RB good_JJ ._.
Battery      not_RB good_JJ

Feature: BATTERY
Opinion: not good
Result : -3.2853896617889404   (-) NEGATIVE Feature
*****

POS tagged text: Battery_NNP becomes_VBZ hot_JJ ._.
Battery      becomes_VBZ hot_JJ

Feature: BATTERY
Opinion: becomesZ hot
Result : 3.048417568206787     (+) POSITIVE Feature
*****
```

```

Java - FeatureBasedSentimentMining/src/FeatureAnalysis.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Console
<terminated> FeatureAnalysis [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 1:41:37 PM)
POS tagged text: The_DT touch_NN works_VBZ nice_JJ ._.
touch          works_VBZ nice_JJ

Feature: TOUCH
Opinion: worksZ nice
Result : 3.5506396293640137      (+) POSITIVE Feature
*****

POS tagged text: The_DT cover_NN is_VBZ not_RB pleasant_JJ ._.
--none--      is_VBZ not_RB

Feature: --NONE--
Opinion: is not
Result : 2.802877187728882      (+) POSITIVE Feature
*****

POS tagged text: The_DT Buttons_NNPS not_RB smooth_JJ ._.
Buttons       not_RB smooth_JJ

Feature: BUTTONS
Opinion: not smooth
Result : -3.456378221511841     (-) NEGATIVE Feature

```

```

Java - FeatureBasedSentimentMining/src/FeatureAnalysis.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Console
<terminated> FeatureAnalysis [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 1:41:37 PM)
*****

POS tagged text: The_DT cover_NN is_VBZ not_RB pleasant_JJ ._.
--none--      is_VBZ not_RB

Feature: --NONE--
Opinion: is not
Result : 2.802877187728882      (+) POSITIVE Feature
*****

POS tagged text: The_DT Buttons_NNPS not_RB smooth_JJ ._.
Buttons       not_RB smooth_JJ

Feature: BUTTONS
Opinion: not smooth
Result : -3.456378221511841     (-) NEGATIVE Feature

Average Semantic Orientation = 1.5174551010131836

(+) POSITIVE Review!!!
*****

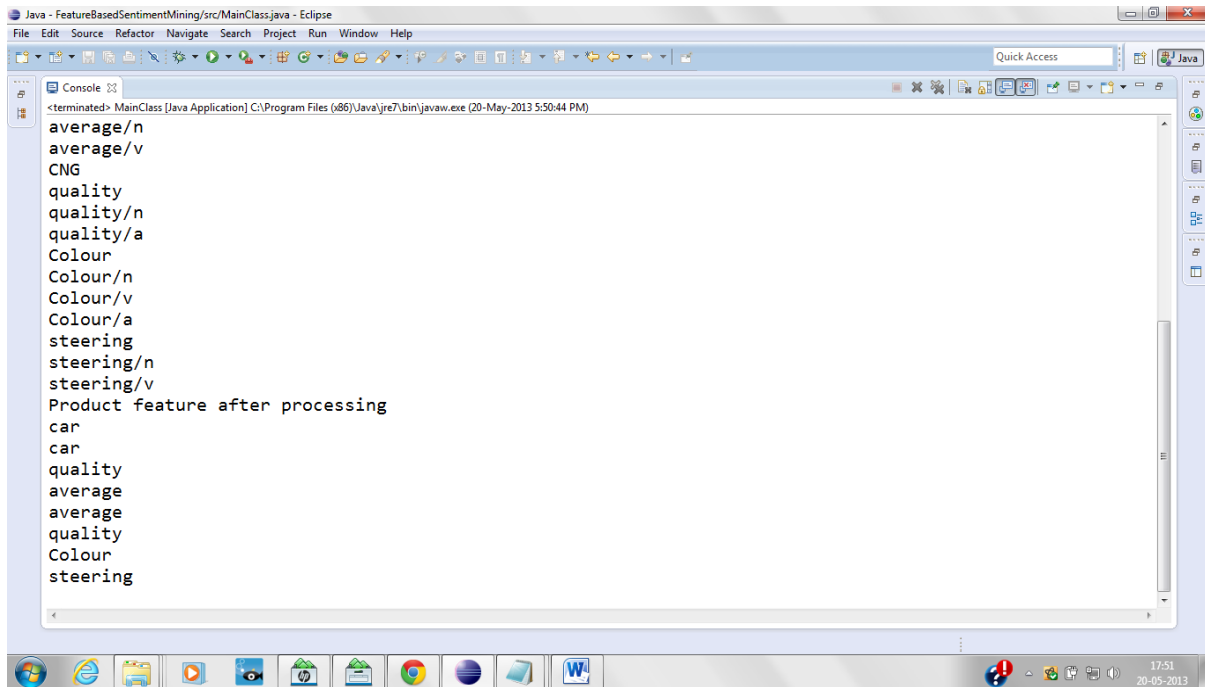
```

Figure 6.5 Semantic orientation

SAMPLE INPUT 2:

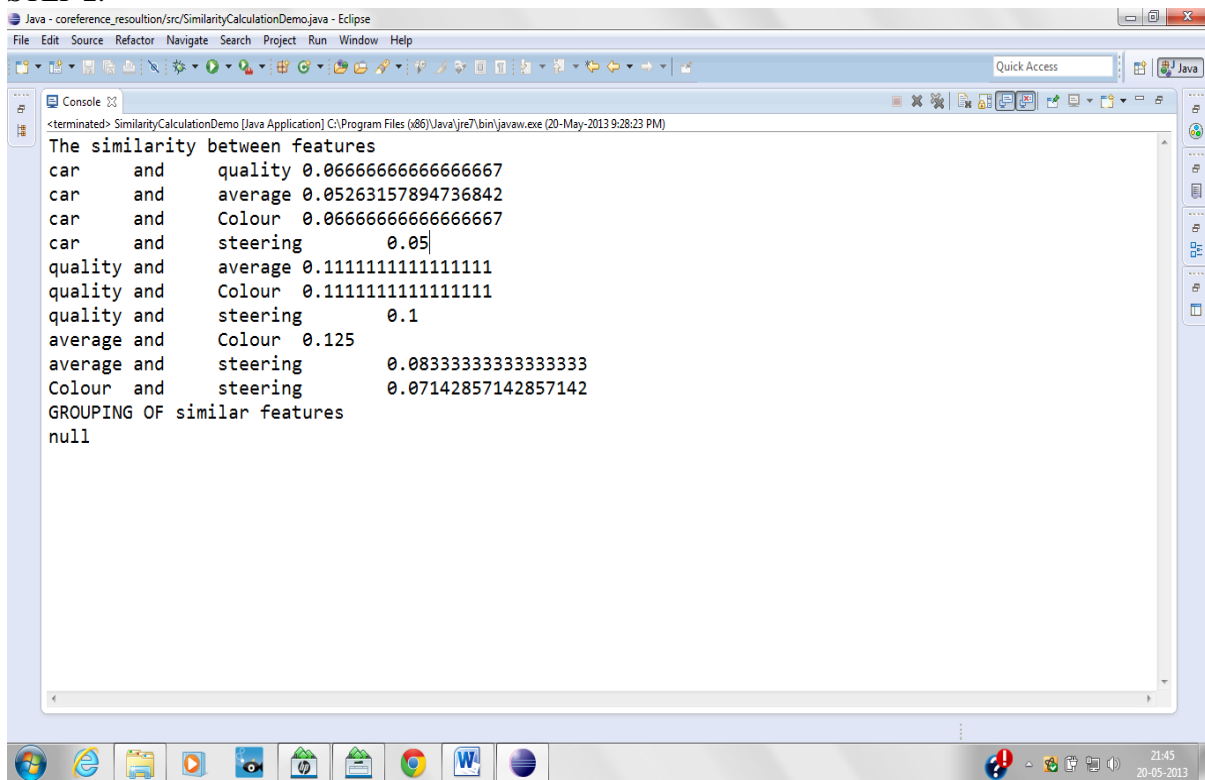
The car looks great. Its quality is strong. Its average is not good. It has high average in CNG. The Interior quality is very good. Colour is not good. It gets rusted quickly. The steering is not good.

Step 1:



```
Java - FeatureBasedSentimentMining/src/MainClass.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> MainClass [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 5:50:44 PM)
average/n
average/v
CNG
quality
quality/n
quality/a
Colour
Colour/n
Colour/v
Colour/a
steering
steering/n
steering/v
Product feature after processing
car
car
quality
average
average
quality
Colour
steering
```

STEP 2:



```
Java - coreference_resolution/src/SimilarityCalculationDemo.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> SimilarityCalculationDemo [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 9:28:23 PM)
The similarity between features
car and quality 0.06666666666666667
car and average 0.05263157894736842
car and Colour 0.06666666666666667
car and steering 0.05
quality and average 0.11111111111111111
quality and Colour 0.11111111111111111
quality and steering 0.1
average and Colour 0.125
average and steering 0.08333333333333333
Colour and steering 0.07142857142857142
GROUPING OF similar features
null
```

STEP 3:

```

Java - coreference_resolution/src/StanfordCoreNlpDemo.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java

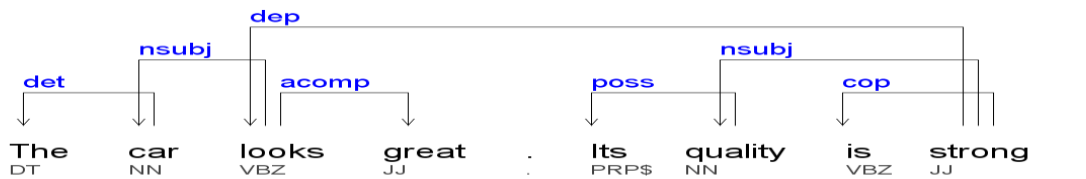
Console
<terminated> StanfordCoreNlpDemo [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 5:56:31 PM)

Sentence #8 (5 tokens):
The steering not good.
[Text=The CharacterOffsetBegin=169 CharacterOffsetEnd=172 PartOfSpeech=DT Lemma=the NamedEntityTag=0] [Te
(ROOT
  (NP
    (NP (DT The))
    (VP (VBG steering)
      (ADJP (RB not) (JJ good))))
    (. .)))

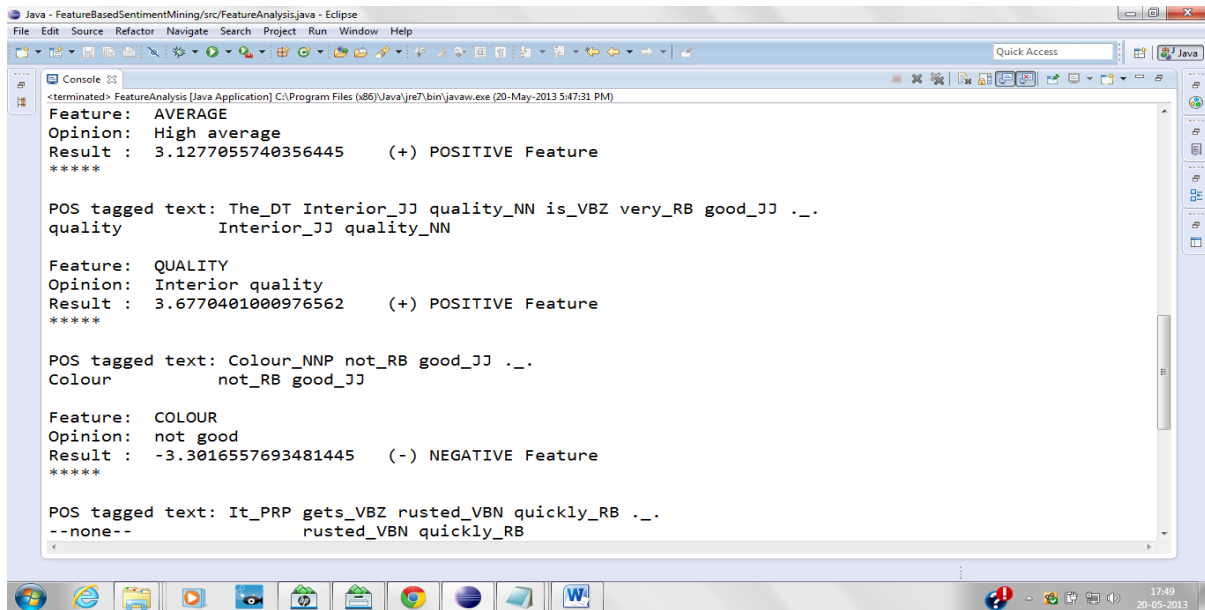
root(ROOT-0, The-1)
partmod(The-1, steering-2)
neg(good-4, not-3)
acomp(steering-2, good-4)

Coreference set:
(2,1,[1,2]) -> (1,2,[1,3]), that is: "Its" -> "The car"
Coreference set:
(3,1,[1,2]) -> (2,2,[1,3]), that is: "Its" -> "Its quality"
Coreference set:
(7,1,[1,2]) -> (6,1,[1,2]), that is: "It" -> "Colour"
  
```

STEP 4:



STEP 5:



```
Java - FeatureBasedSentimentMining/src/FeatureAnalysis.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help

Console
<terminated> FeatureAnalysis [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 5:47:31 PM)
Feature: AVERAGE
Opinion: High average
Result : 3.1277055740356445 (+) POSITIVE Feature
*****

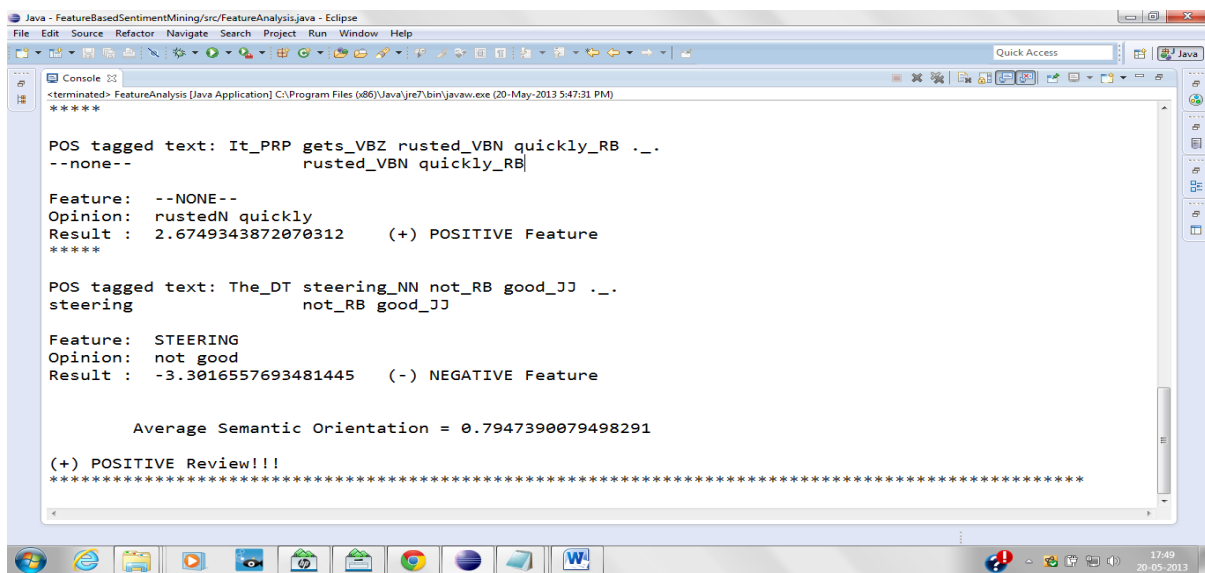
POS tagged text: The_DT Interior_JJ quality_NN is_VBZ very_RB good_JJ ._.
quality Interior_JJ quality_NN

Feature: QUALITY
Opinion: Interior quality
Result : 3.6770401000976562 (+) POSITIVE Feature
*****

POS tagged text: Colour_NNP not_RB good_JJ ._.
Colour not_RB good_JJ

Feature: COLOUR
Opinion: not good
Result : -3.3016557693481445 (-) NEGATIVE Feature
*****

POS tagged text: It_PRP gets_VBZ rusted_VBN quickly_RB ._.
--none-- rusted_VBN quickly_RB
```



```
Java - FeatureBasedSentimentMining/src/FeatureAnalysis.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help

Console
<terminated> FeatureAnalysis [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (20-May-2013 5:47:31 PM)
*****

POS tagged text: It_PRP gets_VBZ rusted_VBN quickly_RB ._.
--none-- rusted_VBN quickly_RB

Feature: --NONE--
Opinion: rustedN quickly
Result : 2.6749343872070312 (+) POSITIVE Feature
*****

POS tagged text: The_DT steering_NN not_RB good_JJ ._.
steering not_RB good_JJ

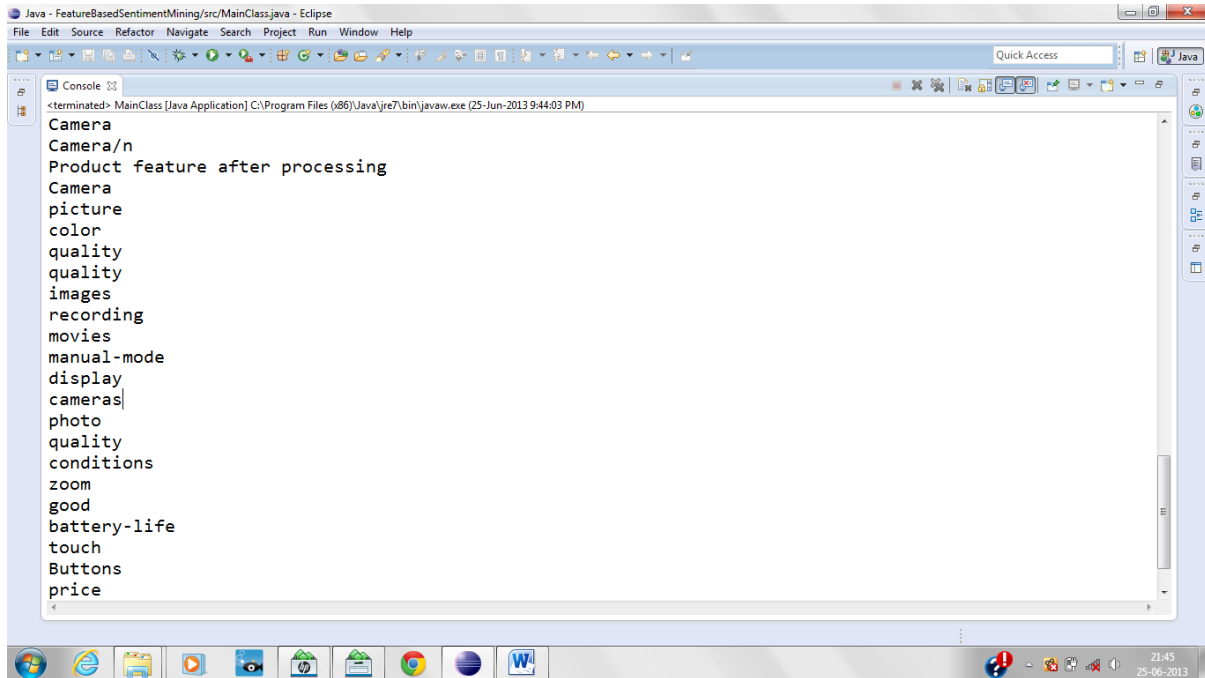
Feature: STEERING
Opinion: not good
Result : -3.3016557693481445 (-) NEGATIVE Feature

Average Semantic Orientation = 0.7947390079498291
(+) POSITIVE Review!!!
*****
```

SAMPLE INPUT 3: CANON (CAMERA)

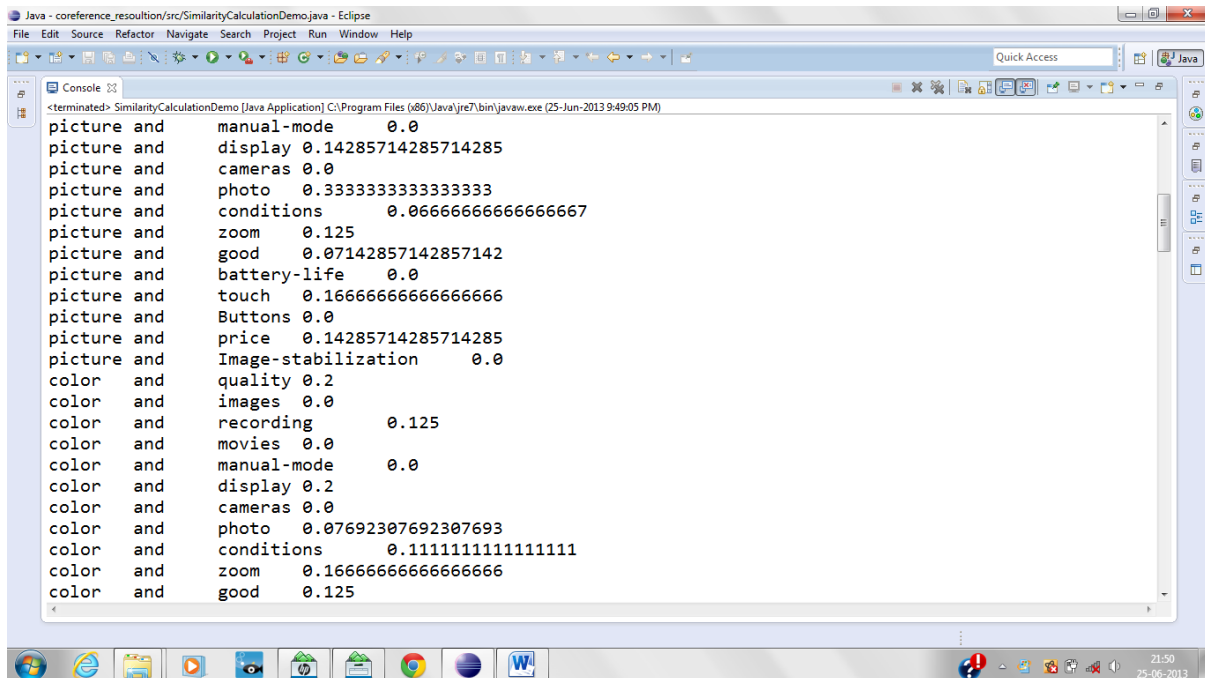
Camera has excellent picture and color quality. It can store high quality images and recording movies. It's not easy to use in manual-mode. The display is not good. Its cheaper than other cameras. It clicks photo of good quality in dark conditions. Its zoom is good. The battery-life is not good. The touch works nice. The Buttons are not smooth. Its price is cheap. The Image-stabilization is not available. Camera is not easy to carry.

STEP 1:



```
Java - FeatureBasedSentimentMining/src/MainClass.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> MainClass [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (25-Jun-2013 9:44:03 PM)
Camera
Camera/n
Product feature after processing
Camera
picture
color
quality
quality
images
recording
movies
manual-mode
display
cameras|
photo
quality
conditions
zoom
good
battery-life
touch
Buttons
price
```

STEP 2:



```
Java - coreference_resolution/src/SimilarityCalculationDemo.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> SimilarityCalculationDemo [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (25-Jun-2013 9:49:05 PM)
picture and manual-mode 0.0
picture and display 0.14285714285714285
picture and cameras 0.0
picture and photo 0.3333333333333333
picture and conditions 0.06666666666666667
picture and zoom 0.125
picture and good 0.07142857142857142
picture and battery-life 0.0
picture and touch 0.16666666666666666
picture and Buttons 0.0
picture and price 0.14285714285714285
picture and Image-stabilization 0.0
color and quality 0.2
color and images 0.0
color and recording 0.125
color and movies 0.0
color and manual-mode 0.0
color and display 0.2
color and cameras 0.0
color and photo 0.07692307692307693
color and conditions 0.11111111111111111
color and zoom 0.16666666666666666
color and good 0.125
```

```

Java - coreference_resolution/src/SimilarityCalculationDemo.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> SimilarityCalculationDemo [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (25-Jun-2013 9:49:05 PM)
cameras and Image-stabilization 0.0
photo and conditions 0.06666666666666667
photo and zoom 0.0625
photo and good 0.07142857142857142
photo and battery-life 0.0
photo and touch 0.07142857142857142
photo and Buttons 0.0
photo and price 0.07142857142857142
photo and Image-stabilization 0.0
conditions and zoom 0.07142857142857142
conditions and good 0.1
conditions and battery-life 0.0
conditions and touch 0.08333333333333333
conditions and Buttons 0.0
conditions and price 0.1
conditions and Image-stabilization 0.0
zoom and good 0.07692307692307693
zoom and battery-life 0.0
zoom and touch 0.2
zoom and Buttons 0.0
zoom and price 0.16666666666666666
zoom and Image-stabilization 0.0
good and battery-life 0.0

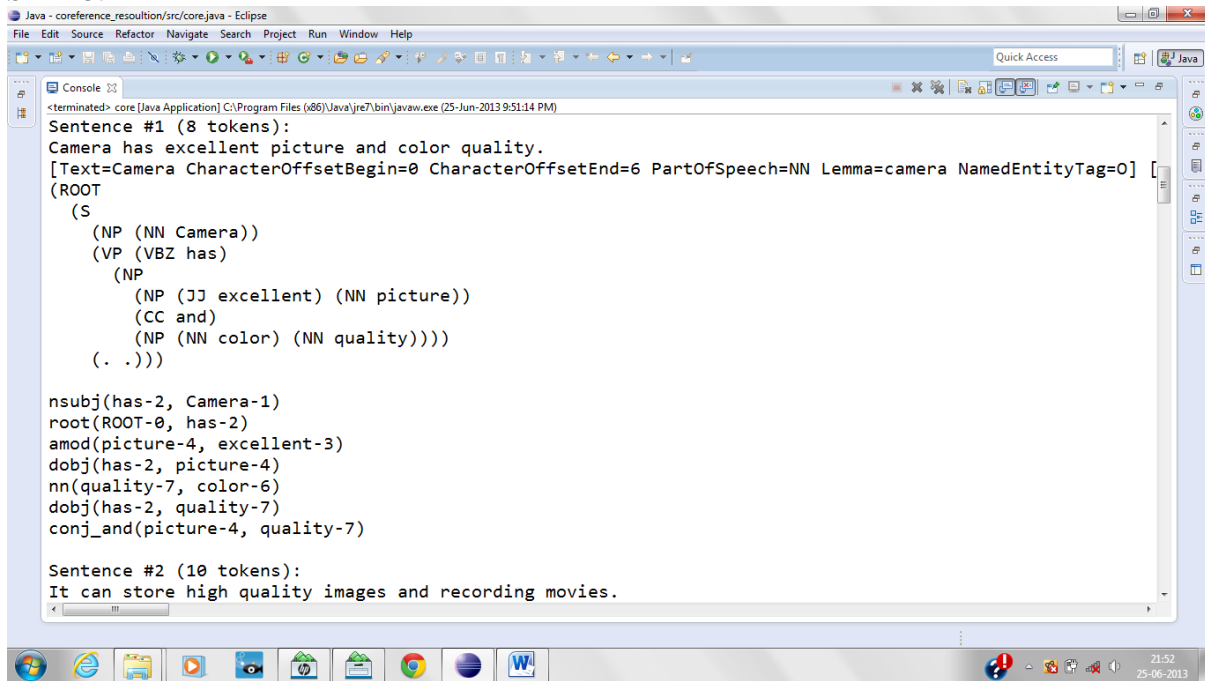
```

```

Java - coreference_resolution/src/SimilarityCalculationDemo.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> SimilarityCalculationDemo [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (25-Jun-2013 9:49:05 PM)
zoom and battery-life 0.0
zoom and touch 0.2
zoom and Buttons 0.0
zoom and price 0.16666666666666666
zoom and Image-stabilization 0.0
good and battery-life 0.0
good and touch 0.09090909090909091
good and Buttons 0.0
good and price 0.14285714285714285
good and Image-stabilization 0.0
battery-life and touch 0.0
battery-life and Buttons 0.0
battery-life and price 0.0
battery-life and Image-stabilization 0.0
touch and Buttons 0.0
touch and price 0.25
touch and Image-stabilization 0.0
Buttons and price 0.0
Buttons and Image-stabilization 0.0
price and Image-stabilization 0.0
GROUPING OF similar features
picture;photo

```

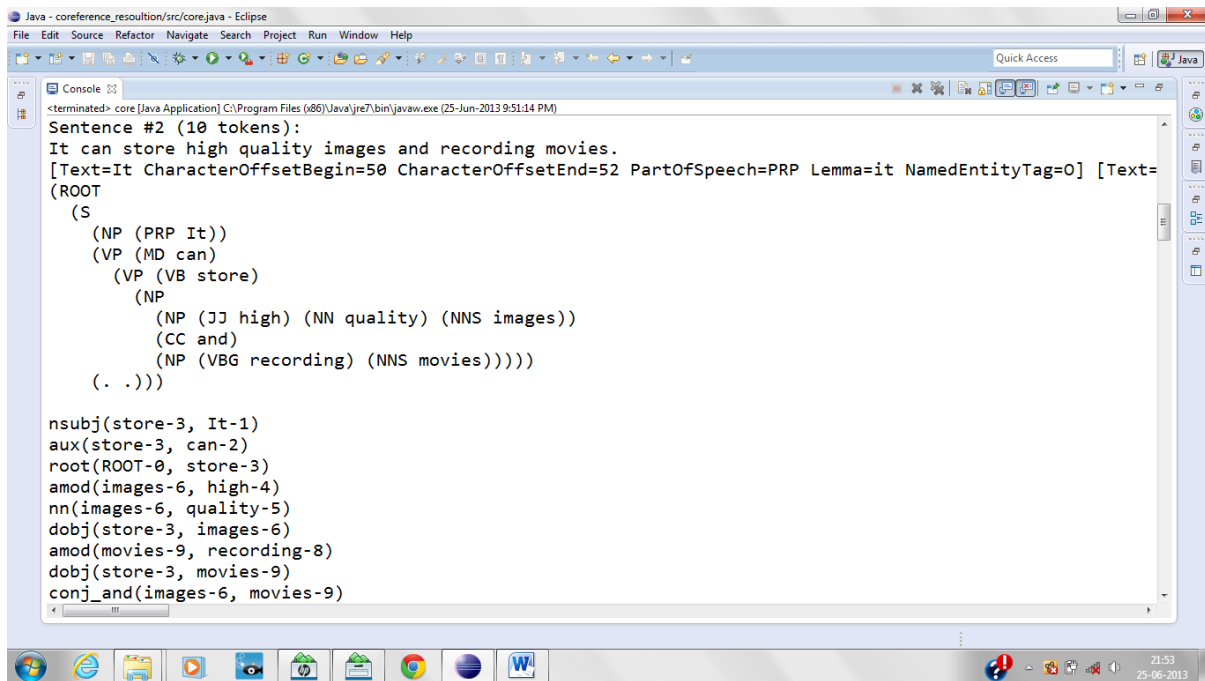
STEP 3:



```
Java - coreference_resolution/src/core.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated>_core [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (25-Jun-2013 9:51:14 PM)
Sentence #1 (8 tokens):
Camera has excellent picture and color quality.
[Text=Camera CharacterOffsetBegin=0 CharacterOffsetEnd=6 PartOfSpeech=NN Lemma=camera NamedEntityTag=0] [
(ROOT
(S
(NP (NN Camera))
(VP (VBZ has)
(NP
(NP (JJ excellent) (NN picture))
(CC and)
(NP (NN color) (NN quality))))
(. .)))

nsubj(has-2, Camera-1)
root(ROOT-0, has-2)
amod(picture-4, excellent-3)
dobj(has-2, picture-4)
nn(quality-7, color-6)
dobj(has-2, quality-7)
conj_and(picture-4, quality-7)

Sentence #2 (10 tokens):
It can store high quality images and recording movies.
```



```
Java - coreference_resolution/src/core.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated>_core [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (25-Jun-2013 9:51:14 PM)
Sentence #2 (10 tokens):
It can store high quality images and recording movies.
[Text=It CharacterOffsetBegin=50 CharacterOffsetEnd=52 PartOfSpeech=PRP Lemma=it NamedEntityTag=0] [Text=
(ROOT
(S
(NP (PRP It))
(VP (MD can)
(VP (VB store)
(NP
(NP (JJ high) (NN quality) (NNS images))
(CC and)
(NP (VBG recording) (NNS movies))))
(. .)))

nsubj(store-3, It-1)
aux(store-3, can-2)
root(ROOT-0, store-3)
amod(images-6, high-4)
nn(images-6, quality-5)
dobj(store-3, images-6)
amod(movies-9, recording-8)
dobj(store-3, movies-9)
conj_and(images-6, movies-9)
```

```

Java - coreference_resolution/src/core.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> core [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (25-Jun-2013 9:51:14 PM)
prep_in(use-5, manual-mode-7)

Sentence #4 (6 tokens):
The display is not good.
[Text=The CharacterOffsetBegin=143 CharacterOffsetEnd=146 PartOfSpeech=DT Lemma=the NamedEntityTag=0] [Te
(ROOT
(S
(NP (DT The) (NN display))
(VP (VBZ is) (RB not)
(ADJP (JJ good)))
(. .)))

det(display-2, The-1)
nsubj(good-5, display-2)
cop(good-5, is-3)
neg(good-5, not-4)
root(ROOT-0, good-5)

Sentence #5 (6 tokens):
Its cheaper than other cameras.
[Text=Its CharacterOffsetBegin=170 CharacterOffsetEnd=173 PartOfSpeech=PRP$ Lemma=its NamedEntityTag=0] [
(ROOT
(S

```

```

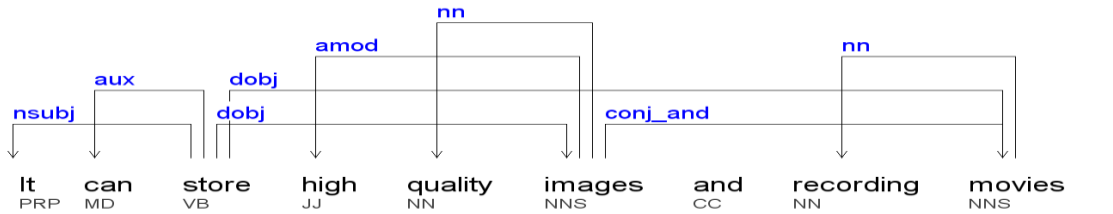
Java - coreference_resolution/src/core.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> core [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (25-Jun-2013 9:51:14 PM)
(S
(VP (TO to)
(VP (VB carry))))))
(. .)))

nsubj(easy-4, Camera-1)
cop(easy-4, is-2)
neg(easy-4, not-3)
root(ROOT-0, easy-4)
aux(carry-6, to-5)
xcomp(easy-4, carry-6)

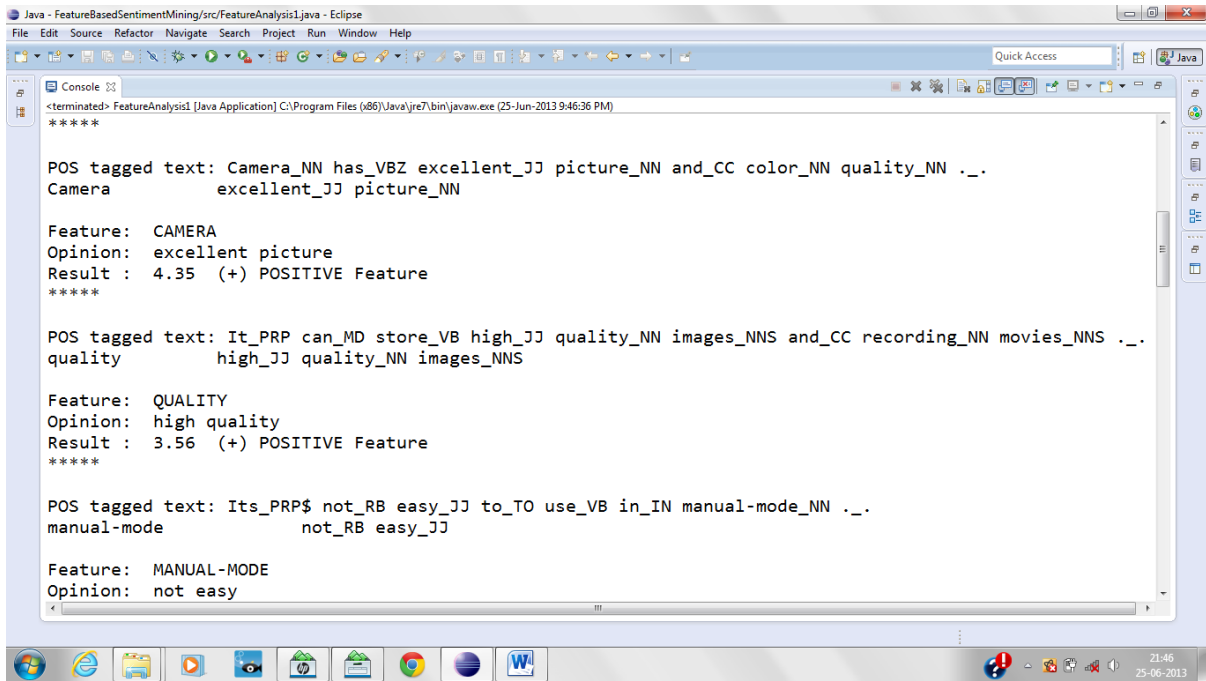
Coreference set:
(2,1,[1,2]) -> (1,7,[6,8]), that is: "It" -> "color quality"
(3,1,[1,2]) -> (1,7,[6,8]), that is: "Its" -> "color quality"
Coreference set:
(5,1,[1,2]) -> (4,2,[1,3]), that is: "Its" -> "The display"
Coreference set:
(6,1,[1,2]) -> (5,2,[1,3]), that is: "It" -> "Its cheaper"
(7,1,[1,2]) -> (5,2,[1,3]), that is: "Its" -> "Its cheaper"
Coreference set:
(11,1,[1,2]) -> (10,2,[1,3]), that is: "Its" -> "The Buttons"

```

STEP 4:



STEP 5:




```
Java - FeatureBasedSentimentMining/src/FeatureAnalysis1.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> FeatureAnalysis1 [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (25-Jun-2013 9:46:36 PM)
*****

POS tagged text: The_DT display_NN not_RB good_JJ ._.
display          not_RB good_JJ

Feature: DISPLAY
Opinion: not good
Result : -2.21 (-) NEGATIVE Feature
*****

POS tagged text: Its_PRP$ cheaper_JJR than_IN other_JJ cameras_NNS ._.
cameras          other_JJ cameras_NNS

Feature: CAMERAS
Opinion: other cameras
Result : 1.1 (+) POSITIVE Feature
*****

POS tagged text: It_PRP clicks_VBZ photo_NN of_IN good_JJ quality_NN in_IN dark_JJ conditions_NNS ._.
photo           good_JJ quality_NN

Feature: PHOTO
Opinion: good quality
```

```
Java - FeatureBasedSentimentMining/src/FeatureAnalysis1.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java
Console
<terminated> FeatureAnalysis1 [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (25-Jun-2013 9:46:36 PM)
*****

POS tagged text: The_DT Buttons_NNPS not_RB smooth_JJ ._.
Buttons          not_RB smooth_JJ

Feature: BUTTONS
Opinion: not smooth
Result : -1.56 (-) NEGATIVE Feature
*****

POS tagged text: Its_PRP$ price_NN cheap_NN ._.
price           price_NN cheap_NN

Feature: PRICE
Opinion: price cheap
Result : 2.14 (+) POSITIVE Feature
*****

POS tagged text: The_DT Image-stabilization_NN not_RB available_JJ ._.
Image-stabilization not_RB available_JJ

Feature: IMAGE-STABILIZATION
Opinion: not available
```

```

Java - FeatureBasedSentimentMining/src/FeatureAnalysis1.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Console
<terminated> FeatureAnalysis1 [Java Application] C:\Program Files (x86)\Java\jre7\bin\javaw.exe (25-Jun-2013 9:46:36 PM)
*****
POS tagged text: The_DT Image-stabilization_NN not_RB available_JJ ._.
Image-stabilization          not_RB available_JJ

Feature:  IMAGE-STABILIZATION
Opinion:  not available
Result :  -2.8 (-) NEGATIVE Feature
*****

POS tagged text: Camera_NN not_RB easy_JJ to_TO carry_VB ._.
Camera          not_RB easy_JJ

Feature:  CAMERA
Opinion:  not easy
Result :  -1.8 (-) NEGATIVE Feature

Average Semantic Orientation = 0.6946153846153847

(+) POSITIVE Review!!!
*****

```

6.2 Evaluation

We now discuss the performance of our system which is analyzed by taking into account the extraction of *feature and opinion* phrases. We calculate the true positive *TP* (total number of correct feature-opinion phrases, the system identifies as correct), the false positive *FP* (total number of incorrect feature-opinion phrases, the system falsely identifies as correct), true negative *TN* (number of incorrect feature-opinion pairs the system identifies as incorrect), and the false negatives *FN* (number of correct feature-opinion pairs the system fails to identify as correct).

We have shown the performance measures for each category of data set. In order to show overall performance of our system, we have done the average of each category. Table 1 summarizes the performance measure values of our system. Since many reviewers do not follow the grammatical rules i.e. a particular sentence structure while writing reviews, therefore we have done pronoun recognition before giving the Semantic orientation of each feature. This technique recognizes more number of features and gives a clear idea of product feature being talked by reviewer. The recall value indicates that some correct feature-opinion pairs are still not recognized by the system correctly. It is calculated using equation 4. The precision value indicates that our system has identified the feature-opinion pairs correctly. It is calculated using equation 3.

By using these values we calculate the following performance measures:

Precision (π): The ratio of true positives among all retrieved instances [21].

$$P = TP / (TP+FP) \quad (3)$$

Recall (ρ): The ratio of true positives among all positive instances [21].

$$R = TP / (TP+FN) \quad (4)$$

F1-measure (F1): The harmonic mean of recall and precision [21]

$$F1 = 2PR / (P+R) \quad (5)$$

Table 1 show that for Canon (camera) product we have recognized 18 true positive (TP) features i.e. it tells the total number of correct feature-opinion pairs the system identifies as correct. Secondly, the false positive (FP) features are 2 i.e. it tells the total number of incorrect feature-opinion pairs the system falsely identifies as correct. After that FN tells false negative features i.e. Total number of correct feature-opinion pairs the system fails to identify as correct are 6. Last, is TN which denotes True negative i.e. total number of incorrect feature-opinion pairs the system identifies as incorrect are 187. After that recall and precision were given using equation (3) and (4).

Product Name	TP	FP	FN	TN	Recall	Precision	F1-measure
Samsung Galaxy (phone)	18	02	06	187	75%	90%	81.81%
Toyata Camry (car)	15	02	04	176	78.94%	78.9%	78.9%
Canon (camera)	14	03	07	195	66.66%	82.35%	73.67%
AVERGAE					73.53%	83.75%	78.12%

TABLE 2 Recall and Precision rate of the system

7.1 Conclusion

Since there is tremendous increase in e-commerce and development of web 2.0 that support actively participation of user, almost every company provides a customer feedback data form on its website. Many sites stress on interaction of users, more and more Websites, feedback forms, such as Amazon, Epinions, UCI lead people to write their opinion about products they are interested in. The users start to express their opinion and they not only use contents passively but also start to create contents actively on blog, web site. Therefore it becomes very difficult for manufacturers to analyze every review for analyzing the product. Therefore it becomes impossible for manufacturers to read every review for analyzing the product. As a result, opinion mining research try to extract information from the opinion data grew up.

In our work, we present a system to identify product features using Stanford Tagger. After that, we have found product features and then combined the similar features using Word Net Similarity. Finally, we used syntactic parse tree for pronoun recognition (co-reference resolution) and finds the sentiment polarity or orientation of opinion sentences using pointwise mutual information and gives feature-based summarily.

7.2 Future Work

In our future work, we will make an experiment on our purposed method for improving effectiveness and accuracy We will research on natural language processing technique for analyzing about implied opinion sentence and analyzing about complex sentence. It is because a review may be written in short sentences and it becomes difficult to create and compare syntactic tree. We will take the words with multi-meanings in different domains before finding the similarity.

For example picture and *movie* are synonyms in movie reviews, but they are not synonyms in digital camera reviews as *picture* is more related to *photo* while *movie* refers to *video*.

References

1. Lect. Shital P. Bora, Sboral@gmail.com., Department of Computer Science and Application, —"DATA MINING AND WARE HOUSING"l, 978-1-4244-8679-3/11/\$26.00 ©2011 IEEE.
2. V. Crescenzi, G.Mecca and P. Merialdo, "RoadRunner: Towards automatic data extraction from large web sites" In *VLDB 2005, Proceedings of 27th International Conference on Very Large Data Bases*, Roma, Italy, pages 109–118. Morgan Kaufmann, Sept. 1.
3. Han kamber, "Data Mining Tutorial", Elsevier, 09-Jun-2011
4. Brijendra Singh1, Hemant Kumar Singh , hemantbib@gmail.com, , Department of computer Applications, AzadIET, Lucknow, INDIA —"WEB DATA MINING RESEARCH: A SURVEY"l, 978-1-4244-5967-4/10/\$26.00 ©2010 IEEE
5. Shiqun Yin, YuhuiQiu, JikeGe,"Research and Realization of Text Mining Algorithm on Web",International Conference on Computational Intelligence and Security Workshops,ISBN number 978-0-7695-3073-4, @ 2007 IEEE
6. Bing Liu, " Mining Comparative Sentences and Relations", Department of Computer Science University of Illinois at Chicago 851 South Morgan Street, Chicago, IL 60607-7053 {njindal, liub}@cs.uic.edu
7. J. Srivastava, R. Cooley, M. Deshpande and P-N. Tan. "*Web Usage Mining: Discovery and Applications of usage patterns from Web Data*", SIGKDD Explorations, Vol1, Issue 2, 2000.
8. RakeshAgrawal, Tomasz Imielinski, Arun Swami, —"Mining Association Rules between Sets of Items in Large Databases"l , Proceedings of the 1993 ACM SIGMOD Conference Washington DC, USA, May 2003.
9. R. Kosala, H. Blockeel "Web mining research" A survey. ACM Sigkdd Explorations,2(1):1-15, 2000.
10. Raymond Kosala, HendrikBlockee, "Web Mining Research: A Survey", ACM Sigkdd Explorations Newsletter, June 2000, Volume 2.
11. Margaret H. Dunham, "Data Mining Introductory & Advance Topics", Pearson Education.
12. Miguel Gomes, Zhiguo Gong, "Web Structure Mining: An Introduction", Proceedings of the 2005 IEEE International Conference on Information Acquisition June 27 - July 3, 2005, Hong Kong and Macau, China
13. Show-Jane Yen, Yue-Shi Lee and Min-Chi Hsieh, —"An Efficient Incremental Algorithm for Mining Web Traversal Patterns", Proceedings of the 2005 IEEE International Conference on e-Business Engineering (ICEBE'05) 0-7695-2430-3/05 \$20.00 © 2005 IEEE.
14. "Research on Page Rank and Hyperlink-Induced Topic Search in Web Structure Mining" International Conference on Internet Technology and Applications (iTAP),ISBN:978-1-4244-7253-6,@ 2011 IEEE
15. Cooley, R. Mobasher, B. and Srivastave, J. (1997) "Web Mining: Information and Pattern Discovery on the World Wide Web" In Proceedings of the 9th IEEE International Conference on Tool with Artificial Intelligence

16. B. Liu, "Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data , Data-Centric Systems and Applications", DOI 10.1007/978-3-642-19460-3_3, © Springer-Verlag Berlin Heidelberg 2011
17. Sotiris Kotsiantis, Dimitris Kanellopoulos, —"Association Rules Mining: A Recent Overview"l, GESTS International Transactions on Computer Science and Engineering, Vol.32 (1), 2006, pp. 71-82.
18. Guandong. Xu, Lin Li, Yanchun Zhang, "Web mining and social networking", Springer, 2011
19. Bin Hu, Jingzhi Yan, Xiaowei Li, "Semi-supervised learning for personalized web recommender system", Computing and Informatics, Vol. 29, 2010, 617–627
20. Zhou, D.—Weston, J.—Gretton, A.—Bousquet, O.—Schölkopf, B.: "Ranking on Data Manifolds. In Advances in Neural Information Processing Systems 16, Cambridge, MA, USA.
21. Bing Liu ,Department of Computer Science, " Sentiment Analysis and Subjectivity", Handbook of Natural Language Processing, Second Edition, (editors: N. Indurkha and F. J. Damerau), 2010
22. JiayunGuo, VladoKešelj, and QigangGao "Integrating Web Content Clustering into Web Log Association Rule Mining", Faculty of Computer Science, Dalhousie University, Springer, 2008
23. Wenhao Zhang, HuaXu, Wei Wan, "Weakness Finder: Expert System with application" 39 (2012) 10283-10291
24. Esuli A., 2008. "Automatic Generation of Lexical Resources for Opinion Mining: Models, Algorithms and Applications". Newsletter ACM SIGIR Forum,42(2)
25. Gamgarn Somprasertsri "Mining Feature-Opinion in Online Customer Reviews for Opinion Summarization" *Journal of Universal Computer Science*, vol. 16, no. 6 (2010), 938-955 submitted: 15/9/09, accepted: 4/3/10, appeared: 28/3/10 c J.UCS
26. Bo Pang and Lillian Lee, "Opinion mining and sentiment analysis", Foundations and Trends in Information Retrieval Vol. 2, No 1-2 (2008) 1–135
27. Mingqing Hu and Bing Liu, "Mining and Summarizing Customer Reviews", Department of Computer Science, KDD'04, August 22–25, 2004, Seattle, Washington, USA Copyright 2004 ACM 1-58113-888-1/04/0008
28. Nan Li a, Desheng Dash Wu, "Using text mining and sentiment analysis for online forums hotspot detection and forecast", Decision Support Systems (Elsevier) 48 (2010) 354–368
29. Fang Kong Guodong Zhou Longhua Qian Qiaoming Zhu* "Dependency-driven Anaphoricity Determination for Coreference Resolution" in Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), pages 599–607, Beijing, August 2010
30. Zhan, Loh, Lui..” Gather customer concerns from online product reviews – A text summarization approach”. Expert Systems with Applications 36 (2009) 2107–2115
31. Zhai, Z., Liu, B., Xu, H., & Jia, P. (2011). "Clustering product features for opinion mining". In Proceedings of the fourth ACM international conference on web search and data mining (pp. 347–354). ACM.
32. Pedersen T. "Information Content Measures of Semantic Similarity Perform Better Without Sense"-Tagged Text. in Proceedings of NAACL HLT. 2010
33. Fellbaum C, WordNet: "An electronic lexical database". 1998:MIT press Cambridge, MA.

34. Jiang J and Conrath D. "Semantic similarity based on corpus statistics and lexical taxonomy". in Proceedings of Research in Computational Linguistics. 1997.19–33
35. Lin D. "An information-theoretic definition of similarity". in Proceedings of ICML. 1998.296-304
36. Mullen, T., & Collier, N. (2004). "Sentiment analysis using support vector machines with diverse information sources". In Proceedings of EMNLP (Vol. 4, pp. 412–418).
37. Pang, B., & Lee, L. (2008). "Opinion mining and sentiment analysis". Foundations and Trends in Information Retrieval, 2(1–2), 1–135.
38. Popescu, A., & Etzioni, O. (2005). "Extracting product features and opinions from reviews". In Proceedings of the conference on human language technology and empirical methods in natural language processing (pp. 339–346). Association for Computational Linguistics.
39. Rushdi Saleh, M., Martín-Valdivia, M. T., Montejo-Ráez, A., & Ureña-López, L. A.(2011). Experiments with SVM to classify opinions in different domains. "Expert Systems with Applications", 38(12), 14799–14804.
40. Tang, H., Tan, S., & Cheng, X. (2009). "A survey on sentiment detection of reviews". Expert Systems with Applications, 36(7), 10760–10773.
41. . Yang, D., & Powers, D. (2005). "Measuring semantic similarity in the taxonomy of word net". Proceedings of the twenty-eighth Australasian conference on computer science (Vol. 38, pp. 315–322). Australian Computer Society, Inc..
42. Yu, H., & Hatzivassiloglou, V. (2003). "Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences". Proceedings of the 2003 conference on empirical methods in natural language processing (Vol. 10, pp. 129–136). Association for Computational Linguistics.
43. Giuseppe Carenini, Raymond T. Ng and Ed Zwart, "Extracting Knowledge from Evaluative Text", Computer Science Department University of British Columbia 2366 Main Mall, Vancouver, B.C. Canada V6T 1Z4 {carenini, rng, ez}@cs.ubc.ca.
44. Zhang, Z., Ye, Q., Zhang, Z., & Li, Y. (2011). "Sentiment classification of Internet restaurant reviews written in Cantonese". Expert Systems with Applications, 38(6), 7674–7682.
45. Stanford Tagger Version 1.6. 2008. <http://www.nlp.stanford.edu/software/tagger.shtml>
46. Stanford Parser Version 1.6. 2008. <http://nlp.stanford.edu/software/lex-parser.shtml>
47. Klein, D., Manning, C. D. 2003. "Fast exact inference with a factored model for natural language parsing". In Advances in Neural Information Processing Systems 15(NIPS 2002), MIT Press Cambridge, Ma, 3-10.
48. Agrawal, R., Imielinski, T., Swami, A. 1993. "Mining association rules between sets of items in large databases". In Proceedings of ACM SIGMOD international conference on Management of data(Washington, D.C., May 25 - 28, 1993). ACM Press, New York, NY, 207-216. DOI=<http://doi.acm.org/10.1145/170072>

49. Church, K. W., Hanks, P. 1990. "Word association norms, mutual information, and lexicography". In Computational Linguistics Volume 16, Issue 1(March 1990). MIT Press Cambridge, MA, 22-29. DOI=<http://doi.acm.org/10.1145/89095>
50. Ryu, Won, Kyu, Ung "A Method for Opinion Mining of Product Reviews using Association Rules" ICIS 2009, November 24-26, 2009 Seoul, Korea Copyright © 2009 ACM 978-1-60558-710-3/09/11
51. M. Abulaish, Jahiruddin, MN Doja, T. Ahmad, Feature and "Opinion Mining for Customer Review Summarization", © Springer-Verlag Berlin Heidelberg PReMI LNCS 5909, pp. 219–224, 2009.
52. Hu, M., Liu, B.: "Mining and Summarizing Customer Reviews". In: Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2004), USA, pp. 168–177 (2004)
53. Liu B, Hu M, and Cheng J. Opinion Observer: "Analyzing and Comparing Opinions on the Web". in Proceedings of WWW. 2005.342-351 May 10-14, 2005, Chiba, Japan. ACM 1-59593-046-9/05/0005.