UNIVERSITY OF CALGARY

MOP: Mining OPinion from Customer Reviews

by

Hong Xu

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

CALGARY, ALBERTA

January, 2009

UNIVERSITY OF CALGARY

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled "MOP: Mining OPinion from Customer Reviews" submitted by Hong Xu in partial fulfilment of the requirements for the degree of Master of Science.

Supervisor, Ken Edwin Barker
Department of Computer Science

Reda S. Alhajj
Department of Computer Science

Jonathan Sillito
Department of Computer Science

Behrouz Far
Electrical and Computer Engineering

October 2, 2008

Date

Abstract

The web is increasingly becoming a very large and excellent knowledge source. It contains a wealth of consumer product reviews. Consumer reviews have already had a significant impact on potential consumers' buying decisions. However, analyzing and classifying the opinion orientations of the consumer reviews for a certain product is time-consuming. In this thesis, a novel mining system is proposed, called MOP (Mining OPinion), which is a feature-based opinion mining system. The mining results are a given product's overall positive scores, negative scores; and a list of its frequent and infrequent features with positive, negative or neutral scores where weights of product reviewers' influence have been incorporated to lower bias arising from opinion holders' specific knowledge and particular requirements. In addition, the positive scores of the given product generated by the MOP system are compared with the average product review value posted online to determine if the average product review is trustworthy. We evaluate MOP using datasets from Amazon.com consisting of the related consumer reviews. The experimental results show that the proposed system is useful and outperforms existing methods.

Acknowledgements

My deepest gratitude goes first and foremost to Dr. Kenneth Edwin Barker, my supervisor, for his constant guidance and encouragement. He has walked me to all the stages of the writing of the thesis. This thesis could not be completed without his consistent and illuminating instructions. In my heart, Dr. Barker is not only a wonderful supervisor, but also a person I can always trust and respect.

I would like to express my heartfelt gratitude to my beloved parents, Xu, ZhenZhong and Guo, ZiLing, for their great confidence and selfless support in my life since the day I was born.

My thanks would go to my first reader of this thesis, Liu, Xin.

I would also like to thank all my friends in Calgary of being there with me throughout the good times and the difficult ones

Dedication

Dedicated to my wonderful parents, Xu, ZhenZhong and Guo, ZiLing.

Table of Contents

List of Tables

List of Figures and Illustrations

Chapter One: Introduction

1.1 Motivation

With the rapid expansion of the Internet, the volume of products sold in the Web is dramatically increasing and more and more Internet users feel comfortable purchasing products online. It is common practice for online merchants, such as amazon.com, to ask customers to write reviews for the products. There also exist some dedicated product review sites, such as epinions.com. To trace customer's product satisfaction, it is necessary to enable customers to review or to express their opinions on the products they purchased.

After shopping online, people like to publish their own consumer product attitudes or reviews about the product using the Web. A wealth of consumer opinions about thousands of consumer products are now available [25]. Consumer product reviews are widely recognized to have a significant impact on consumer buying decisions [7].

Opinion mining, also called *sentiment analysis* (SA) [23], is the task of obtaining writer's feelings as expressed in positive or negative comments, questions, and requests by analyzing documents. The problems of opinion mining have received more attention over recent years, and many researchers have studied this area [5, 25, 26, 31, 33, 57].

Opinion mining is very useful and helpful for both product manufacturers and potential customers. First, for product manufacturers, collecting consumer opinions of the products is crucial. The reason is that many product manufacturers need to identify the strength and weakness of their products for use in marketing intelligence and product benchmarking. Second, for potential customers, reading product reviews on web sites is important to make a good decision and reducing consumer dissatisfaction when purchasing is made.

However, analyzing and classifying opinions from product reviews are time consuming for the following reasons:

(1) Each review written by a product reviewer is unedited and highly variable, due to the reviewer's education level and writing skill. Some product reviews do not adequately describe the strength and weakness of the product. Thus, potential consumers, as readers, may find it difficult to comprehend the key elements of the reviews.

(2) Most potential consumers expect to identify the overall evaluation of the product and to examine the specific strengths of its features. Nevertheless, extracting features of certain products is more complicated because product reviews can be very long but only a few sentences contain opinions on particular features that are of concern to potential consumers. This makes it very difficult for potential consumers who want to buy a product to read these reviews and to classify and analyze the opinions for features of particular interest to them.

(3) To avoid creating bias from a small number of product reviews, potential consumers must read a sufficient number of product reviews. Reading the large number of the reviews to synthesize critical elements presents an insurmountable challenge to the average web users.

Techniques are now being developed to undertake opinion mining to help both product manufacturers in their marketing intelligence and potential consumers in their purchasing decision. Much work has been done recently on opinion mining, which provides a large open resource for improving their work.

Some major work focuses on retrieving and summarizing article contents, instead of extracting and analyzing article opinions. The main purpose of this work is to select or rewrite a subset of the original sentences from a single, or from multiple documents, to capture the main points as document summarizations. For example, some research summarizes the documents [13, 15, 50], but does not aggregate the opinions to indicate if the documents are positive or negative.

Some work extracts and identifies the overall opinion from an entire article, instead of focusing on feature-based opinion extraction and analysis. Opinmind[1] and Mishne [36], for instance, are able to extract and analyze the opinions' essence using the temporal

[1] http://opinmind.com

patterns of captured individual sentiments. However, their work cannot extract and analyze features of a certain product.

Product reviews are written by opinion holders, also called product reviewers (as is used in this thesis), so product reviewers have a great impact on product reviews. Product reviewers are different from news reviewers. News article reviewers usually express their opinions or attitudes about social and political issues of government and act on behalf of a group of people, organizations, or countries. By grouping news article reviewers, we can distinguish different stances on diverse social and political issues and differentiate the relationships among groups of people, among organizations, or among countries. For example, "Democrats in Congress accused vice president Dick Cheney's shooting accident" or "Shiite leaders accused Sunnis of a mass killing of Shiites in Madaen, south of Baghdad". Instead of describing a new social event or a government's act in the news articles, product reviewers commit themselves to express attitudes about a certain consumer product. Usually, these attitudes are personal views written by individual product reviewers, but without any social or political agenda. Product reviewers like to narrate a certain consumer product or its specific features. For example, "I love this HP printer", "I like the color of this Sony camera", or "the picture quality of the camera is amazing". Our system is a product opinion mining system with product reviews as its resource. In this thesis, we focus on product reviewers in our system.

However, most work does not adequately consider product reviewers, in that there may be bias for the opinions extracted from the large number of related articles because of the

product reviewers' general education levels, writing skills, specific knowledge about a certain product, particular requirements, *etc.*

To improve the accuracy of opinion mining from product reviews, a novel mining system called MOP (Mining OPinion) is developed. It draws from a wide range of fields spanning data mining, machine learning, natural language processing, statistics, databases, and information retrieval.

In this thesis, the MOP system proposed is a feature-based opinion mining system. Using our system, the users can easily see the positive, negative, and neutral scores of the overall evaluation results of the product, as well as frequent and infrequent features. To be more precise, a product's *frequent features* are those that have been described in a large number of its product reviews. Conversely, *infrequent features* are those that an individual user emphasized, but may not be described in many product reviews. Thus, using the MOP system, the users can obtain not only the overall evaluation of the product, but also the strength and weakness of its frequent and infrequent features.

The affinity is calculated by comparing the positive overall evaluation result of a product generated by the MOP system with its average product review posted online to determine if the average product review is trustworthy. If the result of the product's positive overall evaluation is within the 95% confidence interval of its average product review posted in the Web, then the average product review is trustworthy.

MOP works in three main stages:

Stage 1: Select a product review web site and download product reviews of a certain product considered by a user. Product reviews are stored in a database. The product's frequent and infrequent features are then identified and extracted. The frequent itemset mining (the first step of association rule mining) and a novel natural language processing are employed in the MOP system.

Stage 2: Extract opinion orientations from product reviews. We focus on extracting opinion words from opinion sentences instead of every sentence in the product reviews, where an *opinion word* is an opinionated content with a positive or negative connotation [14] and is primarily used to express subjective opinions [56]. Here, we require a sentence to contain at least one product feature candidate identified in Stage 1 as an *opinion sentence*. Natural language parsing processor (NLParser[2]) and the large English lexical database called (WordNet[3]) are employed in the MOP system. Each opinion sentence in a product review is classified as positive, negative, or neutral.

Stage 3: To lower bias arising from reviewer's specific knowledge or particular requirements, the product vote rating weighted value is utilized in the MOP system. In this thesis, we define the *specific knowledge* as a product reviewer's knowledge level, which is directly related to the consumer product he purchased. We also define the *particular requirements* in that people's individual requirements are fixed by the

---

[2]http://www.infogistics.com/posdemo.htm

consumer product's distinct features. The affinity is calculated by comparing the positive overall evaluation result of a product generated by the MOP system with its average product review posted online to determine if the average product review is trustworthy.

These three main stages in the MOP system have been decomposed into eight steps. The structure of our system is shown in Section 3.1, where the relation of the eight steps is demonstrated. The implementation detail of each step is described in detail in Section 3.2. Our technique is suitable for the two review formats in the Web. First, at the data source, a "pros and cons" format explicitly captures key features. The product reviews can write the positive opinions of the product in the "pros" section and the negative opinions in the "cons" section. C|net.com uses the "pros and cons" format. Second, a "free format" allows a product reviewer to write the product review with free format. Positive and negative opinions are not separated in this format but are mixed together in the reviewer's description. Sites such as amazon.com use the "free format".

For a "pros and cons" format, opinion orientations of the product's features are separated already, since the property of the format is to indicate product reviewers to write positive points of the product in a "pros" section and negative points in a "cons" section. The MOP system extracts the features of a given product with known opinion orientations from each individual product review. It is possible that one product feature appears in the "pros" section of one product review and also appears in the "cons" section of another

---

[3] http://wordnet.princeton.edu

product review because writing product review is an individual behaviour for the product reviewers. Thus, one product reviewer may think the product feature is positive, but another maybe determine to put the same feature to the "cons" section as negative. In the above situation, the MOP system counts the same feature as a positive opinion once and counts it as a negative opinion once.

However, for the "free format", it is more challenging, because all positive and negative opinions are mixed together. When working with free format, we must identify and extract features from each opinion sentence in every entire product review, and then classify and analyze the opinion orientations that can be positive, negative, or neutral.

1.2 Key Contribution

The thesis contributes in at least the following ways:

1. Provides an opinion mining system where only a URL is a required input.
2. Presents the MOP system that is a feature-based opinion mining system.
3. Extends the opinion words to capture adjectives, verbs, adverbs, and nouns.
4. Lowers the bias arising from product reviewers' specific knowledge and particular requirements by setting a vote rating weighted value for each product reviewer.
5. The affinity is calculated by comparing the positive overall evaluation result of a product generated by the MOP system with its average product review posted online to determine if the average product review is trustworthy.

1.3 Thesis Organization

This thesis is organized in five chapters as follows: the Introduction, Chapter 1, a brief Literature survey is presented in Chapter 2 to describe the state of the art in this area. The full design and implementation detail of the MOP system is described in Chapter 3. Experiments are shown in Chapter 4. Finally, Chapter 5 concludes by stating contributions and presenting future works.

Chapter Two: Literature Survey

The MOP system is a feature-based opinion mining system. It involves two main tasks, extracting features and identifying opinion orientations. In this chapter, we describe related work for each task.

In this chapter, Section 2.1 describes the first main task in the MOP system, namely, extracting features. The document retrieve as a branch of the information retrieve is explained in Section 2.1.1. The word co-occurrence extraction is reviewed in Section 2.1.2. Using a language model to extract features is discussed in Section 2.1.3. The MOP system utilizes the part of speech tagging to extract features, which is described in Section 2.1.4.

Section 2.2 describes the second main task in the MOP system, identifying opinion orientations. The genre classification is reviewed in Section 2.2.1. Section 2.2.2 describes opinion extraction from blogs, stocks, citations, and movies. In Section 2.2.3, the opinion extraction from product reviews is reviewed.

2.1 Extracting features

2.1.1 Document retrieval

Document retrieval is defined as the matching of some stated user queries against a set of free-text records[4]. These records could be unstructured text, such as newspaper articles, real estate records, or paragraphs in a manual. User queries can range from multi-sentence to a few words. There has been considerable research in the area of document retrieval for over 30 years [2], dominated by the use of statistical methods to automatically match natural language user queries against records. For almost as long there has been interest in using natural language processing to enhance single term matching by adding phrases [15]. Hearst [21] and Sack [45] classified entire documents using models inspired by cognitive linguistics. The above work can target particular types of articles and even utilize perspectives in focusing queries (e.g. filtering or retrieving only editorials in favour of a particular policy decision). Goldstein *et al.* [18] and Salton *et al.* [46] presented the methods about document retrieval. However, they can neither extract the product's features nor identify the opinion orientations of the product or its features required by a feature-based opinion mining system, such as the MOP system. DeJong [13] and Tait [50] emphasized identification and extraction of certain core entities and facts in a document. The framework requires background knowledge to instantiate a template to a suitable level of detail. Document retrieval differs from the

---

[4] http://en.wikipedia.org/wiki/Information_retrieval

work reported here in that it largely motivated by issues associated with information retrieval (IR). This means there is an assumption that the user needs to obtain the summary content of one or more document but is searching for specific information in a more structured document. The final summary is likely to focus on a particular topic and can be assessed via a structured query. This work also tends to start from a clearly defined *lingua* that can be defined *a priori*, thereby facilitating more accurate, specific information retrieval but within a narrower scope. It is different from our work as our techniques do not fill any template and are domain independent.

2.1.2 Word co-occurrence extraction

Keyword extraction is an important technique for document clustering, summarization, text mining, Web page retrieval, *etc.* Matsuo [34] presented a new keyword extraction algorithm, where frequent terms are extracted first, and then a set of co-occurrences between each term and the frequent terms, i.e., occurrences in the same sentences, is generated. Co-occurrence distribution shows the importance of a term in the documents as follows. If the probability distribution of co-occurrence between term α and the frequent terms is biased to a particular subset of frequent terms, then term α is likely to be a keyword. However, Matsuo's work can only apply to a single document and does not address identifying product features.

2.1.3 Language model approach

Scaffidi *et al.* [47] applied a language model approach with the assumption that product features are mentioned more often in a product review. To achieve the approach, they configure their system with statistics on how often each part of speech (POS) appears in generic text. Thus, Scaffidi *et al.* configure the system with part of speech frequency data derived from a 100 million word corpus of spoken and written conversation English. Scaffidi *et al.* score each product on each product's feature in a category. Users can select a category (such as fiction books) and quickly retrieve products which are highly rated on a particular feature of that category (such as "ghost story").

2.1.4 Part of speech tagging

Hu *et al.* [25] proposed the idea of opinion mining and summarization to extract product features by applying part of speech tagging to identify nouns and noun phrases. They used unsupervised itemset mining to extract product features. Like Hu *et al.*'s work, the MOP system also employs an itemset mining tool to identify frequent nouns. However, to improve accuracy of identifying product features, our system involves a feature refining method that is applied to merge the candidate features of the product.

2.2 Identifying opinion orientations

2.2.1 Genre classification

Genre is a heterogeneous classificatory principle, which is based on, among other things, the way the text is created, the way it is distributed, and the register of language it uses [29]. Genre classification groups a set of documents into smaller sets according to some predefined genre classes. Genre classification differs from text classification. Text classification techniques typically use the frequency of terms in the documents to discriminate between documents of different topics. However, genre classification classifies text into different styles, such as novel, news, poem *etc.* Wiebe [56] reported on document level classification, using a k-nearest neighbour algorithm based on the total count of subjective words and phrases within each document. Some similar work [17, 28] for genre classification can recognize documents that express opinions, but they do not tell whether the opinions are positive or negative in their work. Although the above work does not address our opinion classification task of determining what the opinion actually is, the genre classification can help recognize documents that express an opinion.

2.2.2 Opinion extraction from blogs, stocks, citations, and movies

More and more people want to extract and analyze useful information from the specific domains, such as blogs, stocks, citations, movies, *etc.*

Chen *et al.* [12] argued for the applicability of a Latent Semantic Analysis (LSA), which is able to provide a higher relevance for the search. Latent Semantic Analysis (LSA) is used for mining content from blogs. It is an automatic indexing and retrieval technique, which is designed for improved detection of relevant documents on the basis of search

queries. LSA addresses two issues from information retrieval broadly named as synonymy and polysemy. Das and Chen [11] used a manually crafted lexicon in conjunction with several scoring methods to classify stock postings. Piao *et al.* [41] presented a system to identify authors' opinions about the works they cite, such as positive or negative attitudes, or approval or disapproval. Their system is based on existing semantic lexical resource tools to create a network of opinion polarity relations between documents and citations. Similarly, Teuful *et al.* [51] also presented an automatic classification of citation functions.

Pang *et al.* [39] examined several supervised machine learning methods applied to sentiment classification of movie reviews and concluded that machine learning techniques outperform the method based on human-tagged features. Tong [52] generated technology to track online discussions about movies. Messages are classified by looking for specific phrases that indicate the author's sentiment towards the movie such as "good editing", "nice visuals", or "wonderful acting". Each phrase must be manually added to a special lexicon and manually tagged to indicate a positive or negative sentiment. The lexicon is domain dependent (e.g. movies) and must be rebuilt for each new domain. Zhuang *et al.* [59] also presented a system for analyzing movie reviews. Although the above three systems are able to extract opinion orientations that indicate the opinions are positive or negative from movie reviews, they are all domain specific.

2.2.3 Opinion extraction from product reviews

2.2.3.1 Opinion extraction from entire product reviews

Morinaga *et al.* [37] presented a framework for mining product reputations on the Internet. The framework can automatically collect reviewers' opinion about the products from web pages and it uses text mining techniques to obtain the reputation of those products. The framework provided by Morinaga *et al.* is helpful for marketing and customer relationship management, which only focus on the overall opinions or reputations of the product in survey data, instead of the special product features. However, it does not mine product features on which the reviewers have expressed their opinions. Although they do find some frequent phrases indicating reputations, these phrases may not be product features (e.g., "don't work" and "benchmark result").

Dave *et al.* [12] built a tool for sifting through and synthesizing product reviews that can automate the work done by aggregation sites or clipping services. It uses structured reviews for testing and training from some web sites, where each review already has a class, such as thumbs-up and thumbs-down, or some other quantitative or binary ratings. It then uses identifying features and scoring methods from information retrieval to determine whether reviews are positive or negative. The approach proposed in the paper begins by training a classifier using a corpus of self-tagged reviews available from the Web. The system then refines the classifier using the same corpus before applying it to sentences mined from broad web searches. Dave *et al.* showed that the classifiers work very well with test reviews. The classifiers have been employed to classify sentences obtained from Web search results, which were obtained by a search engine using a

product name as the search query. However, the performance is poor because a sentence contains much less information than an entire review [58]. However, a feature-based opinion mining system must identify opinion orientations of the product's features that are described in the product reviews. Thus, a feature-based opinion mining system must be able to extract the opinion orientations from the sentences in the product reviews and perform well at the sentence level.

2.2.3.2 Feature-based opinion extraction

Conrad and Schilder's work [8] identified the author's viewpoint about a specific discussion or topic. Their approach divides this problem into sub-problems, including subjective analysis, polarity analysis, and polarity degree. Classify sentences into positive, negative, or neutral viewpoint. At the sentence level classification, there are two primary types of approaches: corpus-based approach and dictionary-based approach.

The corpus-based approach finds co-occurrence patterns of words to determine the opinion orientations of the words or phrases [20, 54]. Popsecu and Etzioni [42] extracted product features and opinions using language patterns. It is built on the top of the KnowItAll system, hence leveraging the KnowItAll's semantic relationships such as hasA() and IsA() relationships. The work by Mei *et al.* [35] showed a method of extracting sentiments on subtopics of blog articles from any *ad hoc* queries. It uses the topic-sentiment mixture model that is extended from the topic mixture model. It is a

probabilistic model that assumes a probabilistic distribution of topical words, and positive and negative words.

The dictionary-based approach, however, uses synonyms and antonyms in a large lexical database of English to determine the opinion orientation of a given word based on a set of seed opinion words, which have been manually created by system developers. Several systems [1, 16, 30] employ the dictionary-based approach. Hatzivassiloglou and McKeown [19] and Kanayama and Nasukawa [27] both employed the dictionary-based approach to identify the orientations of context dependent opinion words. To improve the accuracy, they use conjunction rules to help classify and analyze the opinion orientation of opinion words. The conjunction rule basically declares that if two opinion words are linked by a conjunction, such as "and", in a sentence, then these two opinion words should be a similar opinion orientation, because the conjunction word, "and", is used to connect grammatically coordinate words, phrases, or clauses. For instance, in the sentence, "the picture is nice and pretty". Even if we do not know the word "pretty", it is possible to speculate that "pretty" is a positive word via the conjunction rule. The reason is that the word "nice" is positive, and a conjunction word "and", connects "nice" and "pretty". Thus, we can presume that "pretty" should have very similar meanings to "nice", so "pretty" is conjectured to be positive by the conjunction rule.

WordNet[5] [43, 44] is a large lexical database of English developed at Princeton University. Nouns, verbs, adjectives, and adverbs are grouped into sets of cognitive synonyms called synsets, each expressing a distinct concept. WordNet presently contains approximately 19,500 adjective word forms, organized into approximately 10,000 synsets. It also currently contains 21,000 verb word forms and approximately 8,400 synsets. Since the special synset structure, some research [16, 25, 33] has employed it to determine opinion orientations.

Yu and Hatzivassiloglou [57] proposed a system that classifies opinions at a sentence level. Their system finds opinion sentences by the similarity of sentences based on shared words, phrases, and WordNet synsets. They determine orientation words by computing the co-occurrences of words with the seed words. They use the fuzzy logic to classify sentiment. Subasic and Huetter [49] manually constructed a lexicon associating words, specifying intensity (strength of affect level) and centrality (degree of relatedness to the category). For instance, the word, "mayhem", belongs to both the violence category and intensity certain level.

Turney [53] presented a simple unsupervised learning algorithm for classifying reviews as recommended or not recommended. The algorithm takes a written review as input and produces a classification as output. First, use a part of speech tag to identify phrases in the input text that contain adjectives or adverbs. Second, estimate the opinion orientation

---

[5] http://wordnet.princeton.edu

of each extracted phrase. A phrase has a positive opinion orientation when it has good associations (e.g. “romantic ambience”) and a negative opinion orientation when it has bad associations (e.g., “horrific events”). Third, assign the given review to a class, recommended or not recommended, based on the average opinion orientation of the phrases extracted from the review. If the average is positive, the prediction is that the review recommends the item it discusses. Otherwise, the prediction is that the item is not recommended.

Pointwise Mutual Information and Information Retrieval (PMI-IR) algorithm is used in Turney' system to estimate the opinion orientation of a phrase and measure the similarity of pairs of words or phrases. The opinion orientation of a word is calculated as the mutual information between this word and the word "excellent" minus the mutual information between this word and the word "poor", where the mutual information is computed by issuing queries to a search engine, AltaVista.

Hatzivassiloglou and McKeown [19] used textual conjunctions. For instance, "fair and legitimate" or "simplistic but well-received" have been separated by similarly and oppositely connoted words. Hatzivassiloglou and McKeown presented a log-linear regression model that uses constraints to predict whether conjoined adjectives are of the same or different orientations, achieving 82% accuracy in their task when each conjunction is considered independently. Combining the constraints across many adjectives, a clustering algorithm separates the adjectives into groups of different orientations and finally, adjectives are labelled positive or negative. Hatzivassiloglou and

McKeown' method has high precision, but a large corpus and a lot of manually tagged training data have been used. In addition, their system works only for adjectives, but not for nouns, verbs, and adverbs.

Hu *et al.* [25] proposed the idea of opinion mining and summarization to extract features of a product and determine whether the reviewer's opinion is positive or negative. To improve the work, Liu *et al.* built Opinion Observer [33]. It allows visualization of opinions on product reviews. The opinions are broken down by components using association rule mining to extract several frequent features. The supervised method and language patterns have been used to accurately extract the features. However, in this work, Liu *et al.* still used only adjectives to determine the opinion orientation, which lowers accuracy. In fact, verbs, adverbs, and nouns are as important as adjectives in determining the opinion orientation. Adjectives, verbs, nouns, and adverbs are opinion words to be extracted and analyzed in the MOP system.

The MOP system is related to these work, but it differs from them. The MOP system focuses on classifying and analyzing opinion orientations on a given product and its features from product reviews. In addition, the MOP system is domain independent. There are no language patterns or corpuses required in our work. To improve the accuracy of the opinion orientation classification, the MOP system identifies the opinion words as adjectives, verbs, adverbs, and nouns to classify the opinion orientations. Our method performs well on the opinion orientation extraction, as is suggested by the testing experience.

2.2.3.3 Holistic Approach to Feature Extraction

Our approach can be thought of as one that draws upon key elements from several different approaches in an attempt to combine the most appropriate tool to attack each of the challenges incrementally. Dang *et al.* [60] work is probably the most closely related work to that reported here. We will see in Chapter 4 that their results are better in terms of precision and recall than ours but our contributes by developing techniques that consider more features. We report here on what aspects they consider in their interesting work and, after we present the details of our approach, demonstrate how we differ from their work.

Deng *et al.* [60] studied the problem of determining the semantic orientations (positive, negative, or neutral) of opinions expressed on product features in the product reviews. They proposed a holistic lexicon-based approach to find opinion words by exploiting external evidences and linguistic conventions of natural language expressions. We believe that this holistic approach is the right one but have extended the set of features to be extracted and ultimately argue that this will produce better results because of its increased scope.

Chapter Three: The Architecture's Modules

This chapter describes the MOP system design in Section 3.1 and detail component descriptions of our system in Section 3.2. The MOP system is decomposed into eight steps. Corresponding to each step in the system design, the detail component description of our system consists of eight sections as follows: Section 3.2.1 describes crawling reviews. In Section 3.2.2, tagging part of speech is represented. Section 3.2.3 explains how identification of frequent and infrequent features is accomplished. In Section 3.2.4, opinion words extraction is described. Section 3.2.5 represents classifying orientation. Refining feature candidates is described in Section 3.2.6 and summary generation is discussed in Section 3.2.7. Finally, the affinity calculation is described in Section 3.2.8.

3.1 System Design

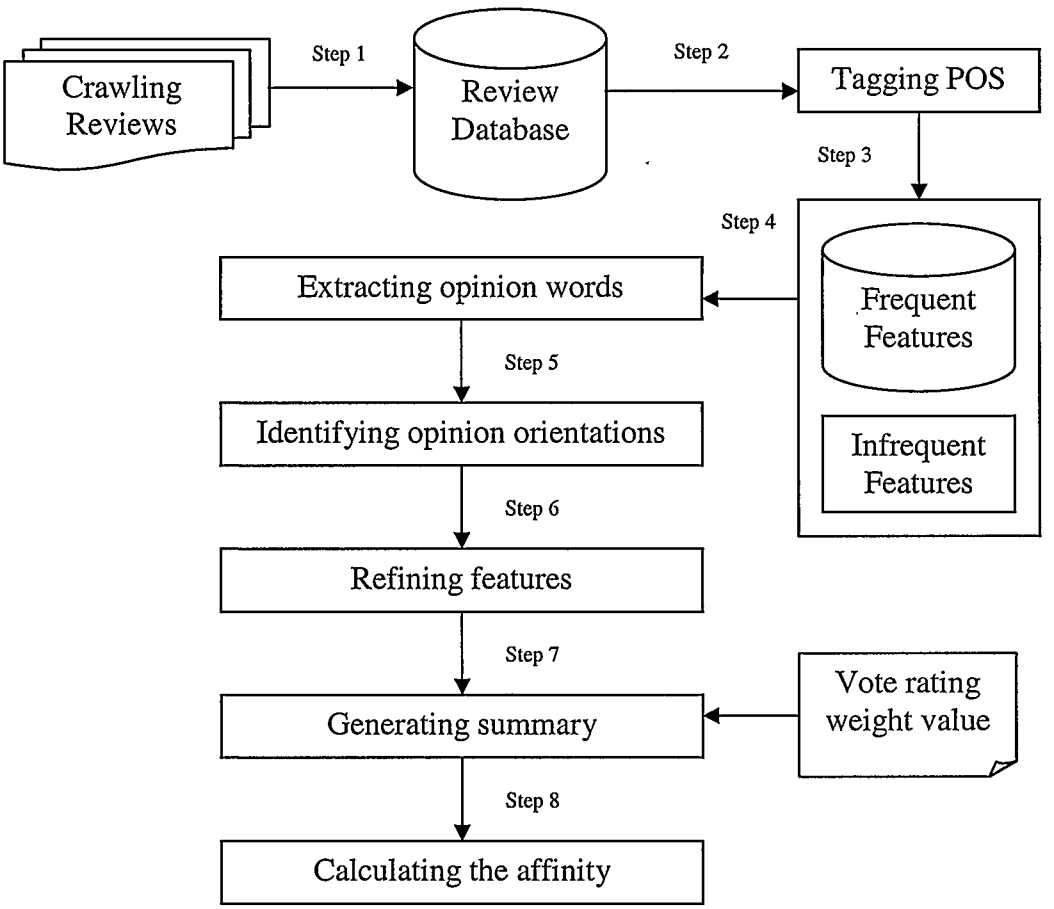


Figure 1: Structure of the MOP system.

Step 1: Certain product's consumer reviews are downloaded and saved in a database

Web Data Extraction (WDE), a web data extractor, is employed in the MOP system for crawling product reviews in the Web. Most product review web sites are structured data web sites. WDE shows high precision and recall for structured data web sites. Thus,

WDE is utilized to extract product reviews from web pages in the MOP system. After crawling, the product reviews of a certain product are saved in the database.

Step 2: Every word in consumer product reviews is processed by part of speech tags

The MOP system applies the part of speech (POS) tagging to identify the part of speech of all words appearing in product reviews. Identifying nouns allows us to find frequent product features; identifying adjectives, verbs, adverbs, and some nouns allows us to classify opinion orientation of a product or its features.

Step 3: Frequent and infrequent features are identified

The MOP system is a feature-base opinion mining system. To identify a certain product's features, we must identify nouns or noun phrases in the product reviews, which can be feature candidates. Infrequent features are collected from the users; and lfrequent features are generated by the frequent itemset mining.

Step 4: Opinion words are extracted from the opinion sentences

To obtain the product reviews' opinion orientations, we must first extract opinion words from the opinion sentences. The Separate into Clauses (SC) method and the Keep Finding Next Clause (KFNC) method are employed to extract opinion words from the opinion sentences in the MOP system.

Step 5: Opinion orientations of opinion sentences are identified

WordNet is utilized in the MOP system to analyze the product reviews' opinion orientations. Although WordNet cannot directly determine if an opinion word is positive or negative, it can indirectly classify a positive attitude or negative attitude of an opinion word based on its synonym by the specific synset structure, in which nouns, verbs, adjectives, and adverbs are grouped into sets of cognitive synonyms.

Step 6: The feature candidates are refined

To describe the product's feature more meaningfully and usefully for users, the MOP system emphasizes a feature refining method. We experientially choose a minimum co-occurrence value as 63%, where we define minimum co-occurrence of all words in a noun phrase as *minimum co-occurrence* in this thesis. If co-occurrence percentage of two feature candidates is equal or greater than 63% in the opinion sentences and their physical positions are next to each other, the two feature candidates are deemed to be one noun phrase feature. We then merge the mining results of the two feature candidates together into one feature.

Step 7: Summary generation is released

The MOP system involves the impact of product reviewers as one of the components in the mining result summary. Utilizing the product review vote rating, the MOP system lowers bias arising from product reviewers' specific knowledge and particular requirements on the product.

Step 8: Affinity is calculated

The affinity is calculated by comparing the positive overall evaluation result of a product generated by the MOP system with its average product review posted online to determine if the average product review is trustworthy. If the result of the product's positive overall evaluation is within the 95% confidence interval of its average product review posted in the Web, then the average product review posted online is trustworthy.

3.2 Detail component description

3.2.1 Crawling reviews

In the MOP system, the first step is to obtain product reviews and save them to a database. A product review is a description of a certain product, usually, in which product reviewers express views, attitudes, or opinions for the product or its features. Two product reviews are shown in Figure 2.

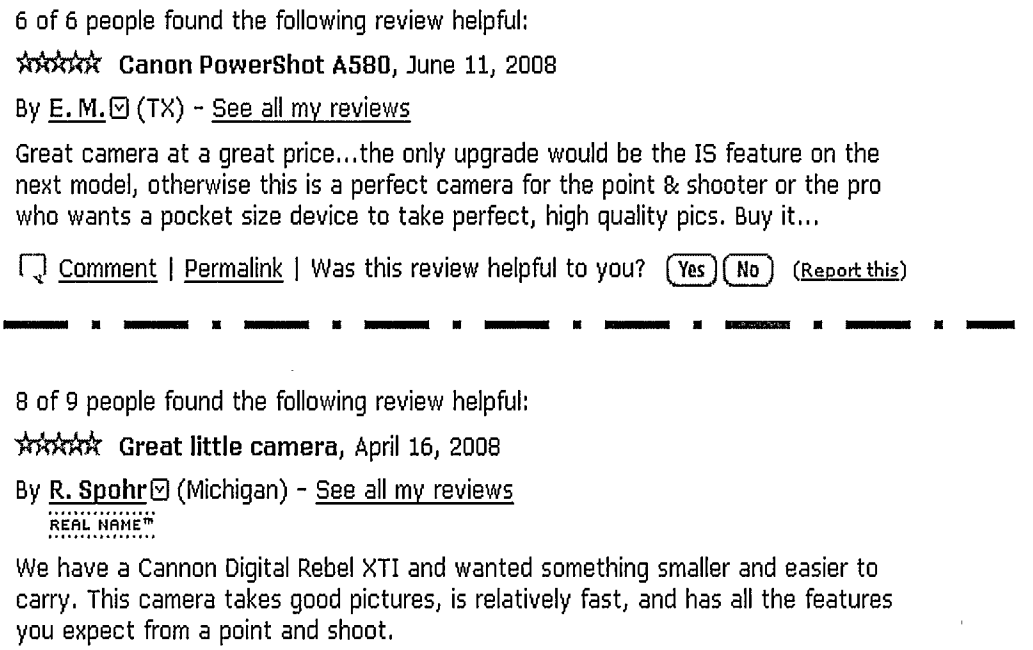


Figure 2: Two product reviews.

The product reviews are saved in a database because it can provide quick and easy ways to access the product reviews for various users' requirement. For example, users are able to select different product reviews as the data source depending on their interests. For example, users can conveniently choose the product reviews according to the time the review posted or the product category. The second benefit to this approach is the DBMS itself is able to manage both the data and its queries in a standard way. In effect, we are able to leverage all of the core DBMS features such as archiving, transactions, and query processing while ensuring that the data itself is normalized and accessible in the most efficient way possible.

To capture the product reviewers' opinion orientations of a product, the first step is to crawl product review web sites and obtain product reviews of a certain product of interest to a user. Structured data web sites have a large collection of pages with an inherent structure or defined schema. These pages are typically generated dynamically from an underlying source such as a relational database, and present them to users through clearly defined data records and presentation patterns. A large amount of information in the Web is presented in regularly structured objects called *data records* [18]. A list of such objects in a web page often describes a list of similar items, such as a list of products or product reviews. Presentation patterns represent a set of data records.

Clearly, product review pages, such as Amazon in Figure 3, are structured data web sites. Since these web pages encode data from a schema, the data encoding is done in a consistent manner across all product review pages. The MOP system focuses on this kind of structured data web sites.

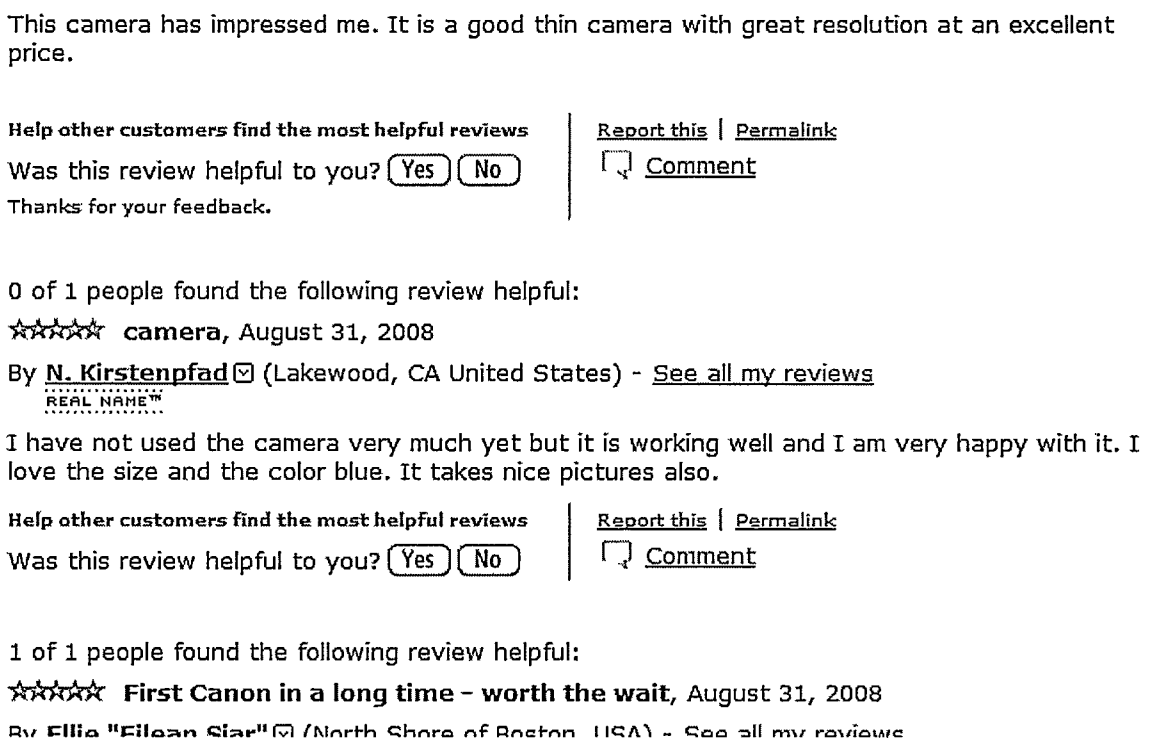


Figure 3: Structured data web site.

Extracting structured data has been recognized as an important problem in information integration systems, which integrate the data present in different web sites. There have been several recent research efforts [15, 53] that address the problem of extracting structured data from web pages, sometimes, called the information extraction (IE) problem.

This thesis studies the problem of automatically extracting structured data encoded in a given collection of pages, without any human input such as manually generated rules or training data. Unlike those work that presents novel algorithms to improve technique to extract the record data, the goal of the MOP system is to modify methods for creating an efficient tool to extract product reviews.

A general purpose web data extractor, called Web Data Extraction (WDE) [40] is available. Compared with MDR [32], the state of the art in web data extraction, with standard notions of precision and recall, WDE shows high precision and record for structured data web sites. *Precision* is defined here as the percentage of the returned data records that are correct; and *recall* is defined as the percentage of the intended data records that are retrieved by the tool. Thus, in this thesis, WDE has been employed to extract product reviews in the web pages.

We define a product review in the crawling review web sites that is saved in the database called a *product review unit* after applying WDE.

3.2.2 Tagging Part of Speech (POS)

In the MOP system, the second step is to produce the part of speech tag for every word occurring in product reviews saved in the database. Our system is a feature-based opinion mining system, so we must extract the product's features and identify the opinion orientations of the product. It is well known that both frequent and infrequent features should be nouns (e.g. quality, color, weight, size, *etc.*) or noun phrases (e.g. SIM card, *etc.*) [33]. Thus, identifying the noun and the noun phrase from other part of speeches is important. Usually, the opinion words can be adjectives (good, bad), verbs (like, hate), adverbs (well, poorly), or nouns (problem). To identify features and opinion words, the first step is to tag the part of speech to all words appearing in the product reviews found

in the database. In the MOP system, the NLProcessor parser has been employed to complete this step.

3.2.2.1 NLProcessor description

NLProcessor is a set of Natural Language Processing technologies developed at the University of Edinburgh in 1990. NLProcessor is able to handle text processing routines, such as tokenization, capitalized word normalization, sentence segmentation, part of speech tagging and syntactic chunking, which are necessary steps in building many kinds of text handling applications.

Normally, an electronic text consists of a sequence of characters including content characters as well as control and formatting characters. Content characters contain letters of an alphabet, numbers, punctuations, *etc.*; and control and formatting characters contain whitespace and newlines. The real text is segmented into linguistic units such as words, punctuation, numbers, *etc.*, before any text processing occurs. This process is called *tokenization*, where the parts of speech are segmented into units called *wordtokens*. Tokenization and wordtokens are utilized in the MOP system because we need to identify the part of speech of words appearing in product reviews. Identifying nouns allows us to find the frequent features of the product. Identifying adjectives, verbs, adverbs, and nouns allows us to classify the opinion orientations of the product or its features.

In the MOP system, we utilize a "capitalized word" feature of NLProcessor to normalize tokens. Sometimes, capitalized words are denoted as proper meanings in the consumer product reviews by reviewers. For example, "the picture quality of this camera is really GREAT". In the sentence, "GREAT" has been capitalized to highlight the picture quality. Here, capitalization presents an emotional emphasis. Capitalized word normalization means that the semantic meanings of the word are lost, so we are able to obtain the normal opinion word and identify the opinion orientation of the word.

3.2.2.2 NLProcessor output

The MOP system uses the NLProcessor's XML output format because it can be easily parsed by DOM Tree[6] later. After being processed by NLProcessor, each product review is marked by some special symbols in an XML file as follows.

```
<P><S><W C = 'DT'> The </W> <W C = 'NN'> battery </W> <W C = 'VBZ'>
is </W> <W C = 'RB'> Very </W> <W C = 'JJ'> good </W> <W T = ','> ,
</W> <W C = 'CC'> but </W> <W C = 'DT'> the </W> <W C = 'NN'> zoom
</W> <W C = 'VBZ'> is </W> <W C = 'JJ'> bad </W> <W T ='.'> .
</W></S></P>
```

Figure 4: NLProcessor's XML output.

[6] http://www.w3schools.com/dom/default.asp

NLProcessor provides special symbols to identify the proper meanings of segment paragraphs, sentences, or words and to tag the part of speech for each word. In

Figure 4, the beginning of each paragraph is marked as <P> and the end of each paragraph is marked as </P>. Similarly, the beginning of sentence is marked as <S> and the end of it is marked as </S>; the beginning of every word is marked as <W> and the end of the word is marked as </W>. This structure makes it straightforward to separate each product review into paragraph, sentence or word levels.

Figure 4 also illustrates that NLProcessor is able to assign the parts of speech tags to words. Here, the word "The" is tagged as a determiner and marked as DT. A noun word "battery" is marked as NN. The word "is" is tagged as a verb and marked as VBZ, which presents as third person. In the sentence, the word "good" is classified as an adjective and written as JJ, *etc.*

Table 1 is a partial list of tags used in NLProcessor, which are often assigned to words in the MOP system.

Table 1: A list of tags in NLProcessor.

POS Tag	Description	Example
JJ	adjective	sweet
JJP	adjective, proper name	Chinese
JJR	Adjective, comparative	sweeter
JJS	Adjective, superlative	sweeter
RB	Adverb	well
RBR	Adverb, comparative	better
RBS	Adverb, superlative	best
NN	Common noun	Zoom
NNS	Noun plural	Pictures
NNP	Proper noun	Canon
VB	Verb base form	Generate
VBD	Verb past	generated
VBG	Gerund	Generating
VBN	Past participle	Taken
VBP	Verb, present, non-3d	generate
VBZ	Verb, present, 3d	generates
CC	Coordinating conjunction	and
DT	Determiner	the
MD	Modal	will

After tagging POS to all words appearing in the product views, the MOP system can easily identify and collect the words with the same type of POS, throughout the corpus. For example, if it is required to identify adjective words, then the only thing we need to do is to collect all words with special symbols, JJ, which means the POS is an adjective.

3.2.3 Identifying frequent and infrequent features

The third step in the MOP system is to identify frequent and infrequent features. Our system is a feature-based opinion mining system, so identifying features, namely, identifying a noun and a noun phrase, is crucial in our work. In this section, we describe how to identify product features; in other words, how to classify a noun and a noun phrase that are product features.

Although the different product reviewers may write or express different aspects of a certain product in their personal attitudes in the Web, most product reviewers like to describe the features that they think are very important. For example, when writing camera reviews, lens, price, picture, zoom, size are mentioned often by camera reviewers [46]. Some popular features appear in the product reviews often, which means many product reviewers think the popular features are important to be mentioned in the reviews. It is possible that the popular features are also highly considered by the potential product buyers. These popular features are called frequent features in this thesis. Finding frequent features in the MOP system makes it appropriate to use frequent itemset mining,

which is the first step of association rule mining, since frequent features exist in most product reviews. Its aim matches the principle of frequent itemset mining.

3.2.3.1 All words

Processing the frequent itemset mining in product reviews requires several key word features, such as ID, NounID, word, and POS. Table 2 shows the detailed description.

Table 2: Features of a word.

ID	An identifier for every different word
NounID	An identifier only for noun word, and none will be signed to other words except noun words
word	The real word
POS	The part of speech of this word

After applying NLProcessor in the MOP system, each product review unit becomes an XML file, where every single word has been tagged by a special symbol that indicates a proper part of speech (POS), and the meanings of all special symbols occurring in our system are listed in Table 1.

To access all words in one product review unit, we need to parse the XML file using DOM Tree. To get all words in all product review units, all XML files must be parsed.

Not all words appearing in product reviews need to be saved in the MOP system. English has a lot of nouns that are always singular or plural [6, 38]. In the English language, nouns that can take plural are called countable nouns [9]. For example, the noun word "apple" is a countable noun, and its plural is "apples". There is also irregular spelling in plural. For example, the plural of "tooth" is "teeth", and the plural of "woman" is "women". Humans can easily recognize some words as singular (e.g. apple, tooth, woman) and others are plural (e.g. apples, teeth, women) even with the different spellings but they express the absolutely same semantic meanings in the product reviews.

However, the computer program NLProcessor assumes that two words are different if their spellings vary. The disadvantage of ignoring the problem is that it lowers the accuracy of frequent itemset mining caused by the inaccurate input, where both the singular and the plural exist. To solve this problem, before assigning a sequential positive number to a nounID for each noun word, the MOP system employs WordNet to detect if the word is a plural of another word that already has a nounID. If it is, the plural is assigned the same nounID as its singular. If it is not, the MOP system will treat the word as a new noun word, and assign a new nounID to the word.

It is worth mentioning that two identical words with different POS values are counted as two different words in the MOP system. In more specific terms, a word in the English language can belong to different syntactic categories. The word, "book", for example, can be a noun or a verb depending on the different situations in the sentence. For example, in the sentence, "I borrowed a java book from the Toronto public library"; "book" is a noun.

However, in the sentence, "I am going to book an air ticket from Calgary to Beijing for my mom"; "book" is a verb. Therefore, in the MOP system, even the same words with different POS values are seen as two different words that are classified into the different POS groups based on their POS values. In the above example, one "book" is classified into the noun group, but another is classified into the verb group.

3.2.3.2 Noun word classification

Applying frequent itemset mining to identify frequent features requires classifying all noun words appearing in all product reviews. We only need to collect the words, the POS characteristic of which is NN. In addition, to identify each noun word, a sequential positive number from one is assigned to NounID as an identifier. Thus, NounID values are real valuable only for noun words. In other words, the value of NounID should be zero for other words except noun words.

Noun words from one product review unit become a group. All groups of noun words become a dataset, where the number of groups is dependent on the number of product review units in the database crawled by the user in the web sites. Based on the requirements of frequent itemset mining, one noun word is allowed to appear only once in a group, even though it may occur many times in this product review unit. However, the order of each noun word physically appearing in a group is not important. The reason is that the principles of frequent itemset mining are to discover how frequently the items co-occur, so the physical positions of noun words appearing in each group cannot impact

the co-occurring frequency of items. Conversely, it is essential to indicate if a noun word occurs in a group because it directly affects mining results.

3.2.3.3 Frequent feature identification

Frequent itemset mining is employed by the MOP system and based on closure algorithms [10] that are Apriori-like algorithm [22]. The aim of frequent itemset mining is to discover all items that co-occur frequently in a dataset. Thus, given a dataset, frequent itemset mining finds all large itemset that have the support greater than the user-specific minimum support. In the MOP system, the minimum support is experientially chosen as 0.060.

After applying frequent itemset mining, we have at least one large itemset, based on the principle of frequent itemset mining. However, the largest itemset, which contains the largest number of items, is chosen by the MOP system because it includes more items than any other itemsets. It is named targetItemset in our system. Obviously, the targetItemset should contain at least one item. Every item in the targetItemset is considered a frequent feature candidate. However, not every frequent feature candidate could be a frequent feature in the MOP system. We are going to explain how to refine frequent features in Section 3.2.6. Each feature candidate in the targetItemset is saved in a collection called featureCollection with its several corresponding location numbers, where each location number consists of a review file number, a paragraph number, and a sentence number. Each location number is able to exactly locate where the feature

candidate originates from a product review unit. One feature candidate might correspond to more than one location number because the feature candidate occurs in several different product review units.

3.2.3.4 Infrequent features

The frequent features can be identified by frequent itemset mining in the MOP system, but infrequent features may not be identified. The reason is that infrequent features are those an individual user emphasized, but they may not be described in many product reviews often. Hence, if an individual user is interested in some particular features, then the MOP system asks the user to directly input additional features but some of these may not occur in the text that often so these are captured manually and called infrequent features.

During runtime, users can input one or more features into the MOP system separated by commas. If users do not have specific features of interest, this step can be skipped and only frequent features provided by the MOP system are considered. For example, a user wants to buy a Sony camera with light weight and pocket size because she considers the weight and size very much. Thus, she can type the words "weight" and "size" into the MOP system during runtime, when the MOP system prompts her to enter infrequent features.

Infrequent features provided by the user are also stored in featureCollection created in Section 3.2.3.3 together with frequent feature candidates with their several corresponding location numbers, where each location number consists of one review file number, one paragraph number, and one sentence number. Similarly to the frequent features' location numbers, each infrequent feature location number is also able to exactly locate where the infrequent feature originates from a product review unit. One infrequent feature may correspond to more than one location number because the infrequent feature can occur in several different product review units.

Infrequent features are entered by users before executing frequent itemset mining. We still use the last example in the above paragraph. The user enters two features, "weight" and "size" as infrequent features. During runtime, the word "size" is generated as a frequent feature when performing frequent itemset mining. Thus, it is possible that an infrequent feature is a frequent feature.

However, each feature is only allowed to appear once in the featureCollection. If two identical words occur in the featureCollection as an infrequent feature as well as a frequent feature, the MOP system only allows the identical words to appear once. All words in the featureCollection should be unique. Thus, if a feature exists in the featureCollection as an infrequent feature, then the feature cannot be stored in the featureCollection again as a frequent feature, even if it is generated as a frequent feature by frequent itemset mining.

In the featureCollection, each feature candidate has several corresponding location numbers that identifies where the feature candidate originates. Each feature candidate could have several location numbers because it might appear in more than one product review unit. However, the use of the location numbers is explained in Section3.2.4.1.

3.2.4 Extracting opinion words

The fourth step in the MOP system is to extract the opinion words from the opinion sentences.

3.2.4.1 Opinion sentence storage

The opinion words are primarily used to express product reviewers' attitudes, ideas or views. These opinion words are usually verbs (dislike or like), adjectives (good or bad), or adverbs (not). Our task is to extract these opinion words from the opinion sentences that must include at least one identified feature candidate.

To extract opinion words, we must first identify opinion sentences. When Dom Tree is used to parse the XML file format of product reviews generated by NLProcessor, every sentence in the product reviews is saved in a collection called SentenceCollection with a unique sentence location number. Here a sentence location number must consist of a review file number, a paragraph number, and a sentence number.

Each feature candidate has at least one corresponding location number. To find the opinion sentences of each feature candidate, the MOP system checks all location numbers of each feature candidate. For every location number of the feature candidate, if it exactly matches with the review sentence's location number in sentenceCollection, then we save the review sentence in a collection called OpinionSentenceCollection with its sentence location number and the sentence is an opinion sentence.

3.2.4.2 Opinion words

Liu's work [33] limits the opinion words to adjectives, which is insufficient to accurately process the natural language. The opinion sentence can be, for instance, "I dislike the lens of this camera", or "This camera really works well". In these opinion sentences, there are no adjective words appearing, so they are assumed to be neutral in Liu's work, which is incomplete. Obviously, in the first sentence, the verb "dislike" indicates that the whole sentence is negative, while in the second sentence, the adverb "well" shows that the whole sentence has a positive attitude. Thus, both verbs and adverbs are crucial for parsing the natural language because they can indicate the opinion orientation of the sentence.

In addition, the noun word is important for processing natural language. For example, consider the opinion sentence, "the only problem with this camera is the battery". Here, the noun word "problem" is an opinion word, since it indicates that the opinion sentence has a negative attitude about the camera's battery. In our system, we consider adjectives,

verbs, adverbs, and nouns to be opinion words. Thus, opinion words can be adjectives, verbs, adverbs, or nouns in the MOP system because all of them are able to express the reviewers' attitudes or views in consumer product reviews. Therefore, the MOP system improves the accuracy of classifying opinion words by empirical evaluation in Chapter Four.

3.2.4.3 Separate into Clauses method (SC method)

In the English language, it is common that one sentence consists of several clauses. Separating a whole sentence into a few clauses improves the accuracy of opinion orientation. For example, consider the opinion sentence, "The zoom is great but the battery is terrible". We assume the word "battery" as the feature required to identify the opinion orientation. When evaluating this whole opinion sentence as a unit to be processed, we extract opinion words for the feature "battery", resulting in two opinion words, "terrible" and "great". The MOP system is conflicted because either the opinion sentence is negative because of the opinion word "terrible" or positive because of another opinion word "great". However, in the above example, if the whole opinion sentence is broken into two clauses by a conjunctive word "but", then the MOP system can capture the correct opinion word "terrible" for the feature "battery". Since the whole opinion sentence is separated by "but", the second clause, "battery is terrible", is highlighted because the feature word "battery" is only contained in the second clause. Now only one opinion word "terrible" is extracted for the feature "battery" and it indicates the negative attitude for the feature "battery".

To accurately orientate the reviewers' opinions, the method, separating a whole sentence into a few clauses by ",", ".", "and", and "but", has been applied previously [19, 27], so we will exploit the existing work. The symbols ",", ".", "and" and "but" are called *separation symbols.*

The MOP system inherits this method and we call it the Separate into Clauses method (SC method). To improve the method of previous work, we also adds "(" as a separation symbol in our system to separate an opinion sentence into clauses. The reason is that round brackets are punctuation marks used in pairs to set apart or interject text within other text, where other text is a comment or notation[7]. Based on the function of round brackets, "(" can be utilized to separate an opinion sentence into clauses as a separation symbol in the MOP system.

3.2.4.4 Keep Finding Next Clause method (KFNS method)

The SC method is helpful to improve accuracy of extracting the correct opinion orientation of the product's features. However, a disadvantage of the SC method is that the opinion words are separated with their feature words when we break up an opinion sentence into clauses by separation symbols. For example, consider the opinion sentence, "I really love the color, size, and zoom of this camera". We assume that the word "zoom"

---

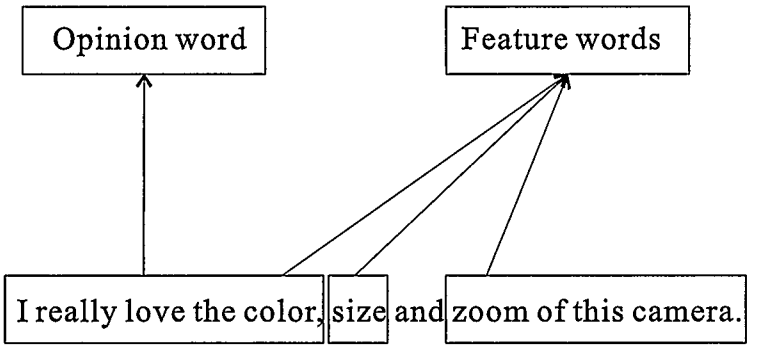
[7] http://en.wikipedia.org/wiki/Round_bracket

is a feature required to identify the opinion orientations. When humans read the whole sentence, they are able to easily recognize that an opinion word "love" in the first clause for the feature "zoom" indicates that the opinion orientation is positive. In the above example, the SC method breaks the opinion sentence into three clauses, "I really love the color", "size", and "and zoom of this camera". Here, the feature word "zoom" is in the third clause, but corresponds to the opinion word "love" in the first clause. If we only apply the SC method in the MOP system, then only the third clause is highlighted. However, there is no opinion word existing in the third clause, so the feature "zoom" is assumed to be neutral. Unfortunately, the parsing result is incorrect.

To solve this problem, a new method called the Keep Finding Next Clause (KFNC) method plays a very important role in the MOP system by finding opinion words and accurately identifying the opinion orientations. The main aim of the KFNC method is that if there is no opinion word in the center clause, we keep looking for opinion words in the next clause until an opinion word is obtained or the last clause is met. We define the clause of an opinion sentence that contains the feature word required to extract as a *center clause.*

The sequence of finding opinion words is described below. First, check if there is any opinion word existing in the center clause. Second, in order, check every clause, the physical position of which is on the left side of the center clause. Third, in order, check every clause, the physical position of which is on the right side of the center clause. The checking process stops until an opinion word is obtained or all clauses of the opinion

sentence have been checked. If no opinion words exist in all clauses of the opinion sentence, then the opinion orientation of the feature is determined neutral in the MOP system. We illustrate the KFNC method in Figure 5.



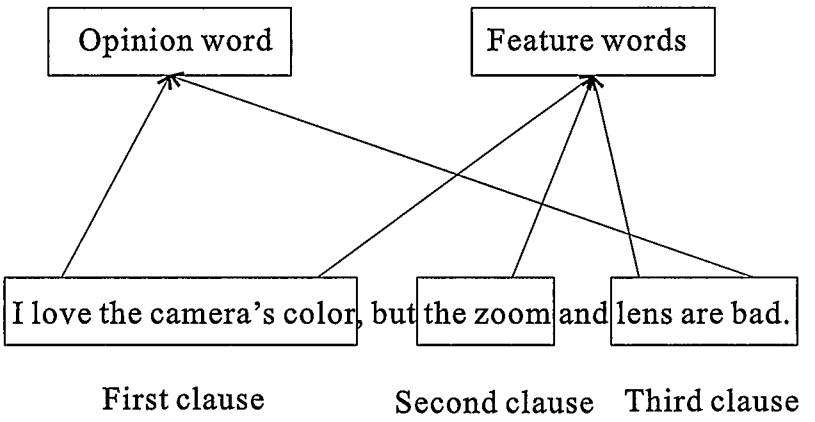
First, the feature word "zoom" is in the third clause that is a center clause, so we check there first. However, there is no any opinion word existing in the third clause.

Second, we keep checking if there is any opinion word in the second clause. However, there is no any opinion word there, either.

Third, we finally check the first clause, and then we found out there is an opinion word "love" occurring there that indicates the opinion orientation of the feature "zoom" is positive.

Figure 5: The KFNC method.

The KFNC method is able to increase the accuracy for identifying the opinion orientations, but it still has a shortcoming. For example, an opinion sentence is "I love the camera's color, but the zoom and lens are bad". We assume the feature "zoom' is required to extract the opinion orientation. The "zoom" is in the second clause, so the second clause is a center clause. The checking process is illustrated in Figure 6.



First, there are no any opinion words existing in the center clause (The center clause is also the second clause).

Second, we keep checking if any opinion word exists in the clause, the physical position of which is on the left side of the center clause. There is an opinion word "like" in the first clause to indicate that the opinion orientation is positive for the feature "zoom". Then, the KFNC method stops because an opinion word is found.

Figure 6: The shortcoming of the KFNC method.

Obviously, the processing result in Figure 6 is wrong. When the humans read the whole sentence, they are able to recognize that the opinion orientation of the feature "zoom" is negative because the opinion word for the feature "zoom" is the adjective "bad" in the third clause, instead of the verb "love" in the first clause. Although there is a shortcoming existing in the KFNC method, it is able to increase the accuracy of identifying the opinion orientations.

3.2.4.5 Order of POS

The MOP system orders POS in three classes. First class: adjective and verb. Second class: noun. Third class: adverb. This order is from strong to weak semantic sentiment classification.

First, check if adjectives and verbs existing in the opinion sentence are able to express an opinion orientation. Opinion words are related to existing work on distinguishing sentences used to express subjective opinions from sentences used to describe some factual information [56]. Previous work on subjectivity [4, 55] has established a positive statistically significant correlation with the presence of adjectives and verbs. The presence of adjective and verbs is useful for predicting whether a sentence is subjective, i.e., expressing an opinion. Thus, the MOP system checks adjectives and verbs first. For example, in a target review sentence, "I like the picture taken by this camera because it helps me easily record a lot of pretty moment in my life." Here, we assume that the word "picture" is a feature word required to identify the opinion orientation. The MOP system applies the SC method first. The opinion sentence is broken into two clauses. The first clause containing the feature word "picture" is highlighted. We determine that the verb "like" indicates the opinion orientation for the "picture" to be positive. The verb "like" is an opinion word and directly and strongly expresses the positive opinion orientation for the feature "picture". Similarly, there is a bicycle opinion sentence, "the color of this bicycle is really nice". Here, we assume "color" is a feature word. The MOP system can directly capture the adjective word "nice" as a positive opinion word.

Second, after checking adjectives and verbs, if the opinion orientation is neutral, then we check if noun words existing in this clause can indicate an opinion orientation. For example, "I love the way this camera dealing with an unexpected problem". The opinion sentence is processed as a unit because there is no separation symbols found in the opinion sentence. Although there is a noun word "problem" to express a negative opinion, a verb "love" indicates that the opinion orientation of the opinion sentence is positive. According to our order of POS, we must check verbs first. The verb "love" can indicate a positive opinion orientation for the opinion sentence. Thus, adjectives and verbs must be checked first. Similarly, there is another opinion sentence, "the only problem of this camera is the battery". Here, we assume "battery" is a feature required to identify the opinion orientation. In the opinion sentence, there are no adjectives or verbs indicating a positive or negative opinion. Consequently, the noun word should be checked. In this example, we find that a noun "problem" is able to express that the attitude towards the opinion sentence is negative.

Third, even after checking noun words, if the opinion orientation is still neutral, then we check if there are any adverbs to indicate a positive or negative opinion orientation of the opinion sentence. The function of an adverb is to modify a verb, adjective, another adverb, or a clause [3]. Benamara *et al.* argued that the adverb affirms an adjective by adverbs of degree and discussed strong intensifying adverbs (e.g. extremely, exceedingly) and minimizer (e.g. hardly). For example, "The concert was hardly good." The adverb "hardly" is a minimizer that reduces the positive degree of the sentence. The minimizer is able to negate the adjective to which they are applied. In the above example, "hardly"

reduces the degree of "good" because good is a positive attitude. The algorithm of extracting opinion words is shown in Figure 7.

```
BEGIN
Initialize l = clause.Length
1  for i = l - 1 to 0 do
2              if ( POS(word[i]) = = adj || verb)
3                     opinion = ClassifyOpinionWordOrientation(word[i])//Figure 9
4                     if (opinion = = positive)
5                            if (word[i]-- = = not || no || n't)
6                                   orientation = negative
7                            else
8                                   orientation = positive
9                     else if (opinion = = negative)
10                           if (word[i]-- = = not || no || n't)
11                                  orientation = positive
12                           else
13                                  orientation = negative
14                    else
15                           orientation = neutral
16 end for
17 if (orientation = = neutral)
18       for i =  l -1 to 0 do
19             if (POS (word[i]) = = noun)
20                           do step 3 to 15
21       end for
22 if (orientation = = neutral)
23     for i =  l to 0 do
24             if (POS (word[i]) = = adverb)
25                    do step 3 to 15
26     end for
END
```

Figure 7: Extracting opinion words.

3.2.5 Classifying orientation

After identifying the opinion words for the corresponding features, we must classify the opinion orientations of the opinion words. Namely, we must identify the opinion words that express the positive or negative attitude. For example, the opinion sentence is "The color of this camera is so great". Here, we assume that the feature word "color" is required to identity the opinion orientation. After extracting the adjective "great" as an opinion word for the feature "color", how can we know that the opinion word "great" is positive or negative?

In the MOP system, we identify opinion orientations based on the synset structure of WordNet. The special synset structure makes it useful for natural language processing. WordNet contains two major classes: descriptive and relational. The MOP system focuses on the former, which is grouped into bipolar clusters. Each cluster is a head synset, in which all adjectives share the same opinion orientation, as illustrated in Figure 8.

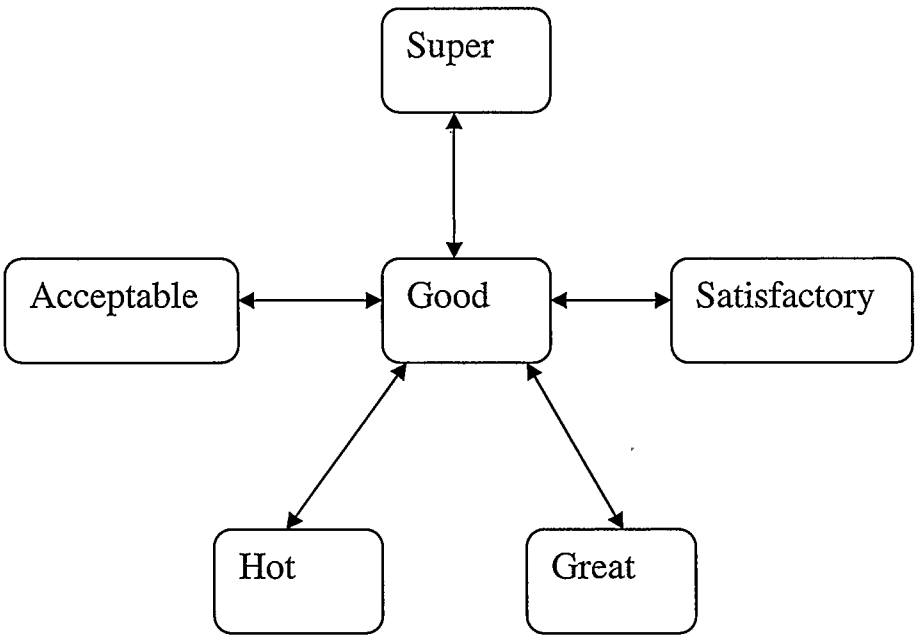


Figure 8: Synset structure of WordNet.

Unfortunately, WordNet cannot directly determine the opinion orientation of any word, which means WordNet is not able to predict a word that is positive or negative. We still use the above example in the last paragraph. In fact, when we extract the adjective "great" as an opinion word for the feature "color" and send it to WordNet, WordNet cannot determine if the opinion word "great" is positive or negative.

However, WordNet has a characteristic that adjectives, verbs, adverbs, and nouns are grouped into sets of cognitive synonyms and share the same orientation as their synonyms in WordNet. According to this characteristic, if a word has a known opinion orientation, then the orientation of its synonym can be set to the same opinion orientation using WordNet. Thus, it is possible to predict the orientation of a verb, adjective, adverb,

or noun by WordNet, if its synonyms' opinion orientation has been known. Thus, if given enough seed verbs, adjectives, adverbs, or nouns with known opinion orientations, then we can predict their synonyms' opinion orientations.

To store some words with known opinion orientations, the primary strategy is to manually create eight seed datasets: PASet, NASet, PVSet, NVSet, PNSet, NNSet, PAVSet and NAVSet in the MOP system. It is explained in Table 3. We manually select 30 common words for each corresponding seed dataset from dictionary.com.

Table 3: Seed datasets.

Dataset	Description	Example
PASet	Positive adjective set	nice, good, great
NASet	Negative adjective set	bad, terrible, awful
PVSet	Positive verb set	love, like, appreciate
NVSet	Negative verb set	dislike, hate, detest
PNSet	Positive noun set	Happiness
NNSet	Negative noun set	problem
PAVSet	Positive adverb set	Better
NAVSet	Negative adverb set	Less

The complete algorithm for classifying opinion word orientation is shown in Figure 9.

We illustrate a procedure to classify the opinion orientation of an opinion word by an adjective. For example, if a given word is an adjective word in the MOP system, then the first step is to check if it is contained in PASet (Positive Adjective Set) or NASet (Negative Adjective Set). If it is, the MOP system identifies the word as a positive or negative adjective based on PASet or NASet.

However, if the given adjective does not exist in PASet or NASet manually created, then it must be sent to WordNet. WordNet provides a synset for this given word. We then check if there is at least one of its synonyms in the synset contained in PASet or NASet. If it is, the given word is classified as a positive or negative adjective according that its synonym is in PASet or NASet. Otherwise, it is assumed to be a neutral adjective.

If the given adjective is classified as a positive or negative adjective using WordNet, then it is added into PASet or NASet based on its orientation. The purpose is to automatically increase the manually created dataset after each performance. The MOP system is able to increase its seed datasets automatically. Therefore, even if a positive adjective is not in the 30 seeds we manually created, it is possible that it can be put into PASet, which can automatically increase PASet.

The process to identify opinion orientations of verbs, adverbs and nouns follows the same procedures as adjectives'.

```
Begin
Initialize W = Opinion Word
1       switch (W) {
2               case adjective:
3                       if (PAset contains W)
4                               return positive
5                       else if (NAset contains W)
6                               return negative
7                       else WordNetChecking(W)
8                       break
9               case verb:
10                      if (PVset contains W)
11                              return positive
12                      else if (NVset contains W)
13                              return negative
14                      else WordNetChecking(W)
15                      break
16              case noun:
17                      if (PNset contains W)
18                              return positive
19                      else if (NNset contains W)
20                              return negative
21                      else WordNetChecking(W)
22                      break
23              case adverb:
24                      if (PAVSet contains W)
25                              return positive
26                      else if (NAVSetset contains W)
27                              return negative
28                      else WordNetChecking(W)
29                      break
30      }
End
```

Figure 9: The pseudo code of classifying opinion word orientations.

```
Begin
1       finding Synset S of W in WordNet
2       for (each synonym s:S){
3               switch (s)
4                       case adjective:
5                               if (PASet contains s)
6                                       add W to PASet
7                                       return positive
8                               else if (NAset contains s)
9                                       add W to NASet
10                                      return negative
11                              else return neutral
11                              break
12                      case verb:
13                              if (PVSet contains s)
14                                      add W to PVSet
15                                      return positive
16                              else if (NVset contains s)
17                                      return negative
19                              else return neutral
18                              break
19                      case noun:
20                              if (PNSet contains s)
21                                      add W to PNSet
22                                      return positive
23                              else if (NNSet contains s)
24                                      add W to NNSet
25                                      return negative
27                              else return neutral
26                              break
27                      case adverb:
28                              if (PAVSet contains s)
29                                      add W to PAVSet
30                                      return positive
31                              else if (NAVSet contains s)
32                                      add W to NAVSet
32                                      return negative
34                              else return neutral
33                              break
36
37      }
End
```

Figure 10: The pseudo code of checking in WordNet.

3.2.6 Refining features

The purpose of refining features is to describe the product's features more helpfully, meaningfully, and usefully for users in the MOP system. Posting product reviews is a personal and individual behaviour in both merchant sites and dedicated product review sites. Thus, some product reviewers might pick different words to express the same or very similar meanings. We emphasize merging features in the MOP system. For example, the "SD card" is one kind of the memory cards for cameras. To save the words, some product reviewers like to write "SD" or "card" instead of the phrase "SD card" in their product reviews. The opinion sentence, for instance, can be "the memory of SD card is not enough", "the capacity of SD is small", or "I do not like the card". The MOP system processes "SD" and "card" as two different feature candidates, but they are actually one noun phrase. The MOP system utilizes the merging feature method to solve this problem.

The feature refining is applied in Step 7, where all the feature candidates have been found and are ready to be released. First, we experientially choose minimum co-occurrence as 63% in the MOP system. Second, we calculate the percentage of two feature candidates, the physical positions of which are next to each other in all opinion sentences containing the symbol "SD", the word "card", or the noun phrase "SD card". Third, we compare the calculation result with minimum co-occurrence. If the calculated result is greater than minimum co-occurrence, then the two feature candidates are deemed to be one noun

phrase feature. We merge the mining results of the two feature candidates into one feature in the MOP system.

3.2.7 Generating summary

In the past work [8, 42, 47, 53], generating the final review summary just involves simple counting of the total number of the positive and negative attitudes from the product reviews. However, their work ignores the impact from product reviewers. The format is shown in Figure 11.

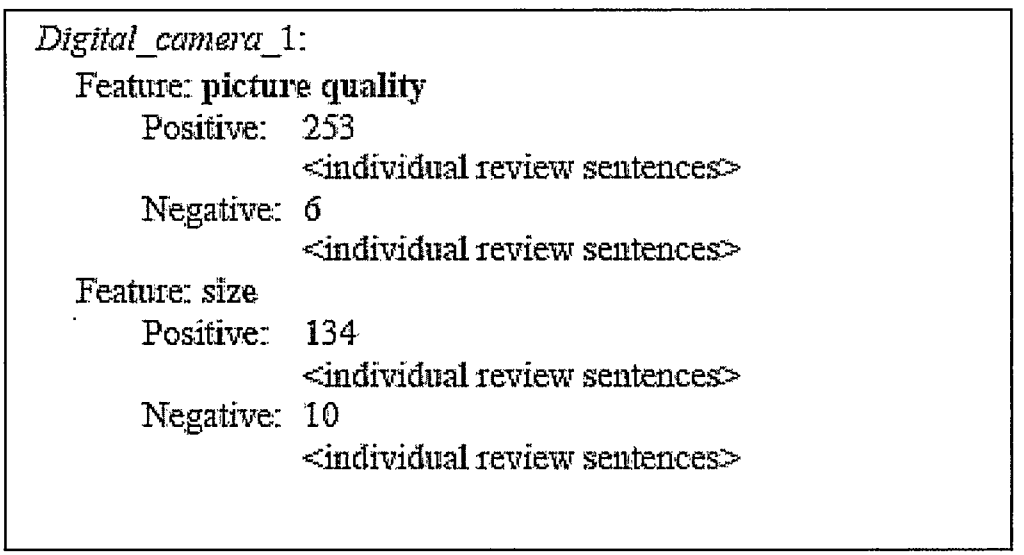


Figure 11: Summary generation.

Adequately considering product reviewers as a component in the opinion mining work is very important. The mining results integrated with the product reviewer's impact are more valuable and trustworthy for the potential consumers.

In fact, the impact of product reviewers is very important in the Opinion Mining system for at least two reasons. First, product reviewers differ from other reviewers, such as news reviewers. Product reviewers only express their personal ideas, attitudes, and reviews for products they purchased. Second, product reviewers describe the opinion orientations related to their specific knowledge and particular requirements. Thus, identifying various product reviewers is very useful for improving the trustworthiness of opinion mining results.

3.2.7.1 Bias from product reviewers

Bias from product reviewers could be multi-faceted. However, in this thesis we consider bias arising from the two aspects: specific knowledge and particular requirements.

3.2.7.1.1 Specific knowledge

The specific knowledge is used to measure how much knowledge a product reviewer has about the product. Product reviewers' specific knowledge might affect conclusions they make because they integrate their individual knowledge into the product reviews posted in the Web. For example, assume a customer holds a PhD in chemistry but has rarely touched video games. When writing a product review for a basic chemistry textbook, he definitely has the authority to criticize the textbook. The reason is his specific chemistry knowledge is tightly related to a basic chemistry text book. However, when writing a

product review for a new video game, he is only a novice because he does not play much. Thus, his specific knowledge about the video game is lacking.

Nowadays, more and more people make use of high-tech products, such as computers and digital cameras, in their lives. However, the consumers lacking electronic knowledge usually like the products to be easily and simply operated. For example, some consumers do not have much knowledge about how to manipulate manual cameras, but like the automatic cameras very much. The reason is that the automatic cameras are able to automatically adjust the focus, change the exposure parameters according to the light, and so on. Thus, this kind of consumers might like to write better product reviews for the automatic cameras than professional cameras.

Consequently, the bias occurs as a result of the product reviewers lacking a specific knowledge about the product. Thus, the product reviewers with dissimilar specific knowledge might express different attitudes or opinions for the products.

3.2.7.1.2 Particular requirements

Particular requirements may affect a product reviewer's attitude towards the product. A particular requirement is a prime reason for purchasing a product. The product can satisfy the users' particular expectation. Some users respect the product with a good quality, but others are sensitive to the product's appearance. For example, some consumers put great importance on the pretty color and portable size when they are looking for a camera.

However, for most professional cameras, the color is black, the size is big, and the weight is heavy. Thus, bias of reviews may occur based on the consumer's requirements because the consumer likes the camera with pretty color and portable size. Thus, this kind of consumers might write poor reviews for the professional cameras.

3.2.7.2 Limitation to lower bias

In the Web, there are some visible limitations to lower bias on products.

First limitation: No specific notation indicates that a product review is from a professional or novice.

Posting product reviews is a highly personal behaviour in merchant sites and dedicated review sites, where no specific notation indicates that a product review is from a professional or novice. Thus, computer programs treat every product review equally, which limits the ability to lower bias from product reviewers.

To resolve the first limitation, we can utilize an additional function that is a product review vote rating. It is provided by many product review sites, such as c|net.com, epinion.com, and amazon.com. For example, in amazon.com as shown in Figure 12, a reader is able to vote on a product review's usefulness by clicking the "Yes" button or indicate it is useless by clicking the "No" button. This additional function can be utilized in the MOP system to reduce bias because the vote ratings represent reader's satisfaction.

If the readers vote the product review as useful, then this product review has at least three advantages. (1) The product review is useful for readers for the time being; (2) The product review includes several topics readers really consider; and (3) The description of this product review should be clear and understandable.

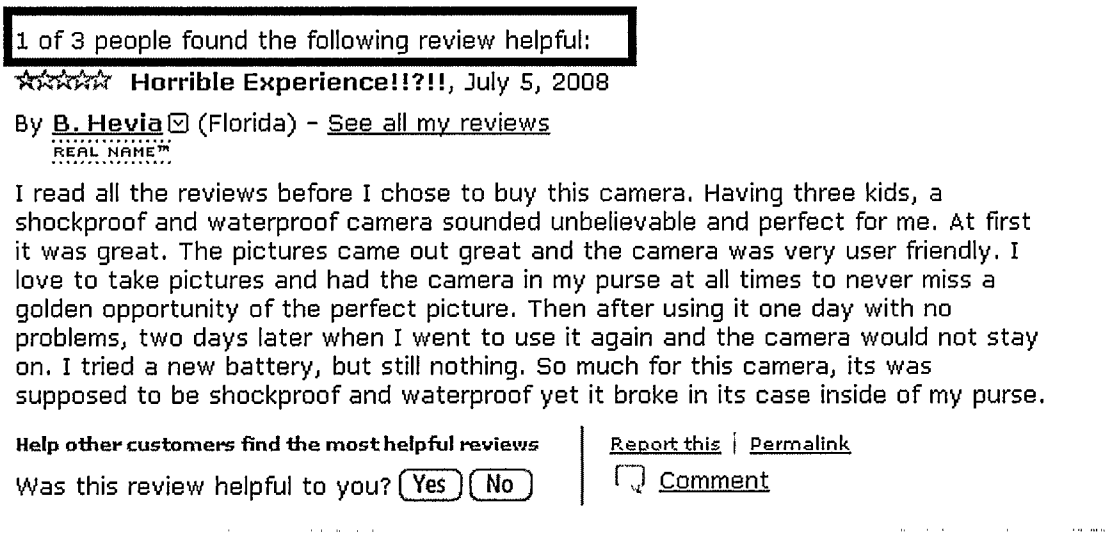


Figure 12: Review vote rating.

Second limitation: Intentional vote, multi-vote, and timeliness impact the result of the review vote rating.

Even though vote ratings can evaluate product reviews, there are still some shortcomings. (1) Intentional vote. Some people spitefully lower or intentionally enhance a product review for some reasons. They may not have even read the entire product review before voting. (2) Multi-vote. Some readers might really prefer or hate a particular product review, so they vote many times against the product review. (3) Timeliness. A product

review might be valuable for a reader when he read it the first time. However, subsequent similar reviews are less valuable for him because the information is being repeated. Thus, the first time reading the product review, he votes for it as useful. Subsequently seeing the same or very similar product reviews might result in votes for uselessness.

To deal with the second limitation, the MOP system assumes that there is no intentional vote in the Web, where any readers only votes once with honesty. This assumption may not reflect reality but is a requirement pragmatically. Techniques to address this are left for future research.

3.2.7.3 Possibility to lower bias

By involving the vote ratings in the MOP system, it is possible to lower bias caused by the product reviewers' specific knowledge and particular requirements.

If a product reviewer has the specific knowledge related to the product he reviewed, then his product review usually shows the following impressions. First, the review has clear and definitive opinions of this product with detailed, reasonable, and practical reasons. Clearly expressing his opinions of this product can convince the potential consumers to buy or not buy the product. Second, the product review is able to point out the characters of the product with explanations. Third, the product review can explain how the product achieves its promise. Fourth, the product review can evaluate the product with the positive or negative attitudes. Usually, the product reviewer with specific knowledge can

evaluate the product and its features and give reasonable reasons. When potential customers read this kind of product reviews, it is more likely that the customers vote them as useful.

If a product reviewer has the particular requirements for the product, then his review may narrow the domain describing the product's features. This indicates that very few features he highly emphasized appear in his product review. The number of readers who vote the product review as useful should be less.

Therefore, it is possible to lower bias from product reviewers' specific knowledge and particular requirements using vote rating.

3.2.7.4 Vote rating weight value calculation

For every review, readers are able to vote the review as a useful review by clicking the "Yes" button. The web sites show the voting result. For example, in amazon.com, the voting result shows "Y (the number of people voting the review as useful) of X (the total number of people voting the review)". If a user has made a helpful vote, then the number of people voting the review as useful increases by one and the total number of people voting the review also increases one. Thus, vote result changes to be "$Y+1$ of $X+1$ people found the following review helpful".

If the voting results show "Y of X people found the following review helpful", we can calculate the number of the unhelpful vote is "$X-Y$". We define that the product review value equals that the number of the helpful vote "Y" subtracts the number of the unhelpful vote "$X-Y$". Thus, the value of each review is "$Y-(X-Y)$". We then compare values of all product reviews and find the maximum value denoted by Max and the minimum value denoted by Min. The helpful value range is the difference between the maximum helpful value and the minimum helpful value. We denote helpful value range as $R=(Max-Min)$.

The impact of each review is calculated as the value of each review $Y-(X-Y)$ divided by the helpful value range $R=(Max-Min)$.

$$\text{VoteRatingWeightValue} = \frac{Y-(X-Y)}{R} = \frac{2Y-X}{R}$$

3.2.8 Affinity calculation

3.2.8.1 Confidence interval for a population mean

The average customer review value is usually denoted by stars in the product review web sites. It is shown in Figure 13. In amazon.com, the highest score is 5 stars and lowest is 1 star. We assume the full score is 100 in the MOP system. Thus, 5 stars are corresponding

to 100, 4 stars to 80, 3 stars to 60, 2 stars to 40, and 1 star to 20, respectively. The average customer review value provides the overall evaluation of the product for potential consumers at a glance.

Figure 13: PalmOne Zire 31 Handheld PDA average customer review.

Due to the different number of product reviewers, the confidence of the average customer review value varies. Obviously, if the average customer review based on 10 product reviewers estimates that the product is good, you may not trust it too much. However, if the average customer review based on 1000 product reviewers estimates the product that is good, you may trust more. Figure 13 shows that 154 product reviewers estimate the product, but we still want to know the average customer review value from all people who purchased the product. Our goal is to estimate the value of an unknown population mean in statistics, where the population mean is the average in "average customer review". Thus, in the MOP system, we calculate the confidence interval for a population mean.

Before introducing the calculation formula of the confidence interval for a population mean, we introduce two concepts related to the population mean computation, confidence interval and confidence level. The *confidence interval* is a formula that tells us how to use sample data to calculate an interval that estimates a population parameter such as a population mean or a proportion. The *confidence coefficient* is the probability that an interval estimator encloses the population parameter, that is, the relative frequency with which the interval estimator encloses the population parameter when the estimator is used repeatedly a very large number of times. The *confidence level* is the confidence coefficient expressed as a percentage [48].

The $100(1-\alpha)\%$ confidence interval for population mean is

$$\overline{x} \pm z_{\alpha/2}\sigma_{\overline{x}} = \overline{x} \pm z_{\alpha/2}\frac{\sigma}{\sqrt{n}}$$

where $z_{\alpha/2}$ is the z value with an area $\alpha/2$ to its right and $\sigma_{\overline{x}} = \sigma/\sqrt{n}$. The parameter σ is the standard deviation of the sampled population and n is the sample size.

Note: When σ is unknown and n is large $(n \geq 30)$, the confidence interval is approximately equal to

$$\overline{x} \pm z_{\alpha/2}(\frac{s}{\sqrt{n}})$$

where s is the sample standard deviation.

The condition is that the sample size n is large $(n \geq 30)$. Due to the Central Limit Theorem, this condition guarantees that the sampling distribution of $\bar{x}$ is approximately normal. The condition is conformable for the MOP system because the MOP system usually deals with a large number of reviews. The number of reviews for most products is more than 30 in amazon.com.

95% is chosen as the confidence level in the MOP system. Most researchers use the 95% confidence level[8], where the 95% confidence level means you can be 95% certain. Commonly used value of $z_{\alpha/2}$ corresponding to 95% confidence level is 1.96. Thus, the 95% confidence interval for population mean in the MOP system is

$$\bar{x} \pm z_{\alpha/2}\sigma_{\bar{x}} = \bar{x} \pm 1.96(\frac{\sigma}{\sqrt{n}}) \approx \bar{x} \pm 1.96(\frac{s}{\sqrt{n}})$$

We illustrate the 95% confidence interval for population mean by the example posted in Figure 13.

In Figure 13, the value of 5 stars is 100, the value of 4 stars is 80, the value of 3 stars is 60, the value of 2 stars is 40 and the value of 1 star is 20. We already assume above values at the beginning of Section 3.2.8.1.

In the MOP system, we calculate the $\bar{x}$ as

---

[8] http://www.surveysystem.com/sscalc.htm#one

$$\bar{x} = \frac{100*58+80*35+60*16+40*12+20*33}{58+35+16+12+33} = \frac{10700}{144} \approx 74.31$$

The sample standard deviation s is

$$s = \sqrt{\frac{1}{N}\sum_{i=1}^{n}(X_i - \bar{X})^2} =$$

$$\sqrt{\frac{58*25.69^2+35*5.69^2+16*(-14.31)^2+12*(-34.31)^2+33*(-54.31)^2}{144}}$$

$$= \sqrt{\frac{154150.30}{144}} = 32.72$$

Therefore, the 95% confidence interval for population mean in the MOP system is

$$\bar{x} \pm 1.96(\frac{s}{\sqrt{n}}) = 74.31 \pm 1.96(\frac{32.72}{\sqrt{144}}) = 74.31 \pm 5.34$$

Then the 95% confidence interval is, approximately from 68.97 to 79.65. That is, at the 95% confidence level, we estimate the value of the average product review for the product to be between 68.97 and 79.65 for the product from all people who purchased the product. Then, the result is involved in the affinity calculation in Section 3.2.8.2. It is compared with the positive scores of the given product generated by the MOP system to determine if the average product review posted online is trustworthy.

3.2.8.2 Affinity calculation

Due to the intentional vote existing in the Web, we calculate the affinity to determine if the value of the average product review is trustworthy. To calculate the affinity, we can check if the positive overall evaluation generated by the MOP system is within the confidence interval. If it is, the value of the average product review is trustworthy; otherwise, it is not trustworthy. The result from the above the example in the last section is used to demonstrate affinity computation. Thus, we know that at the 95% confidence level, we estimate the average product review value for the product to be between 68.97 and 79.65 for the product from all people who purchased the product.

To get the positive overall evaluation for the PalmOne Zire 31 Handheld PDA, we use the PalmOne Zire 31 Handheld PDA customer reviews in Figure 13 as a dataset to run the MOP system. The positive overall evaluation is 75. We then check if 75 is between the confidence interval from 68.97 to 79.65. The answer is positive, so the average product review value posted online is trustworthy in the example.

Chapter Four: Experiments

The MOP system has been implemented using Java. Our system is evaluated from three perspectives: first, the effectiveness of feature extraction; second, the effectiveness of opinion sentence extraction; third, the accuracy of orientation prediction of opinion sentences; and forth, comparison of our system with other holistic approaches.

4.1 Data Sets

Product reviews were downloaded from amazon.com for seven products: four digital cameras, two PDAs, and one road bike. We collected product review data sets shown in Table 4. Each of the product reviews includes a text review, the vote ratings, and the average customer view. Additional information is also available, such as the product review posting date and time, the link to all product reviews posted by the same author, *etc.*, which are not used in the MOP system.

Table 4: Product review dataset.

Dataset	Number of reviews
Canon SD1000	252
Canon A570IS	200
Panasonic Lumix DMC-TZ3A	232
Nikon D40	207
Palm TX handheld PDA	204
PalmOne Zire 31 Handheld PDA	154
GMC Denali Road Bike	148

For each of seven products, we first crawled the product reviews by the Web Data Extraction (WDE) and stored them into the database. NLProcessor was then applied to generate part of speech tags for every word in all product reviews.

4.2 Evaluation of feature extraction

4.2.1 Feature extraction preparation

To evaluate the feature extraction, we manually read all product reviews of seven products, and identified product features and opinion sentences. Table 5 lists the hand-picked set of features that we think are relevant to the digital cameras, PDAs, and road bikes by manually reading product reviews.

Table 5: Expected features.

Expected features		
Digital Camera	PDA	Road Bike
Camera	Weight	Price
Picture or image	Bluetooth	Quality
Zoom	Screen	Seatpost or seat
Battery	SD card	Stem capability
Price	Flash Memory	Handlebar
Quality	headphones	Brake
Shot	battery	Wheels
Pocket or size	price	Derailleur
Screen	Quality	
SD card		

4.2.2 Feature extraction Experiments

The first goal is to obtain the recall and precision of feature extraction. The second goal is to determine how many features can be extracted while varying the support parameters of the closure algorithm for mining frequent items. Third goal is to determine the value of minimum co-occurrence when the candidate features are refined

To achieve the first goal, we calculated the recall and precision of the feature extraction generated by the MOP system. The results are shown in Table 6. During the evaluation, we found that the feature "zoom" was missed the most in 4 camera datasets; the feature "headphones" was missed the most in 2 PDA datasets; and the feature "stem" was missed the most in the road bike. The reason is that very few reviews discuss the zoom, headphones, and stem capability, respectively.

Table 6: Recall and precision of feature extraction in MOP and in Hu and Liu's work.

Dataset	Feature extraction			
	MOP		Hu and Liu's work	
	Recall	Precision	Recall	Precision
Canon A570IS	0.87	0.85	0.80	0.77
Canon SD1000	0.89	0.88	0.83	0.79
Panasonic Lumix DMC-TZ3A	0.86	0.84	0.81	0.80
Nikon D40	0.84	0.83	0.83	0.76
Palm TX handheld PDA	0.84	0.81	0.80	0.75
PalmOne Zire 31 Handheld PDA	0.83	0.81	0.80	0.76
GMC Denali Road Bike	0.79	0.77	0.76	0.75
Average	0.84	0.82	0.80	0.77

Table 6 shows the precision and recall of the features generated by the MOP system. Column 1 lists the product names we used as datasets. Column 2 and Column 3, respectively, list the recall and precision of the features generated by the MOP system. Column 4 and Column 5 show the recall and precision of the features generated by Hu and Liu's work [24].

We compared the features extracted by our system with the features generated by the well known opinion mining system presented by Hu and Liu [24]. Obviously, the average recall and precision of the MOP system are higher than Hu and Liu' work. After analyzing the results from Hu and Liu's work, we found that there are two primary

reasons making their results lower. First, when extracting noun phrases, Hu and Liu utilized a feature provided by NLProcessor that can generate a special tag for the noun phrases. However, this feature provided by NLProcessor does not work very well for product reviews. It creates too many noun phrases and a lot of noun phrases are not the product features. Although Hu and Liu did the pruning after creating features, they cannot remove most noun phrases that are not product features. Second, both the singular and plural of a feature exist in the Hu and Liu's work, but they express the absolutely same meanings. However, in the MOP system, before extracting features, we employ WordNet to eliminate plural and singular discrepancies. It improves the results. From the average of recall and precision, we know that the MOP system is more effective than Hu and Liu's work on extracting the product features.

To achieve the second goal, we varied the support parameter of the closure algorithm from 0.1 to 0.9. Figure 14 shows that how many features can be extracted while varying the support parameter of the closure algorithm for mining frequent items. In Figure 14, the support parameter is on the x-axis and the number of features extracted by the MOP system is on the y-axis, where 150, 200, and 250 reviews are shown, respectively. We chose 15 as the optimal number of features. It seems more than the number of the expected features (around 10) we hand-picked because the MOP system cannot combine some features with some symbols together. For example, for the camera datasets, the MOP system cannot recognize that the symbol "pics" and "pic" represent the feature "picture". Combining these symbols with features is reserved for the future work. Thus,

the number of features extracted by the MOP system may be more than the number of the expected features hand-picked.

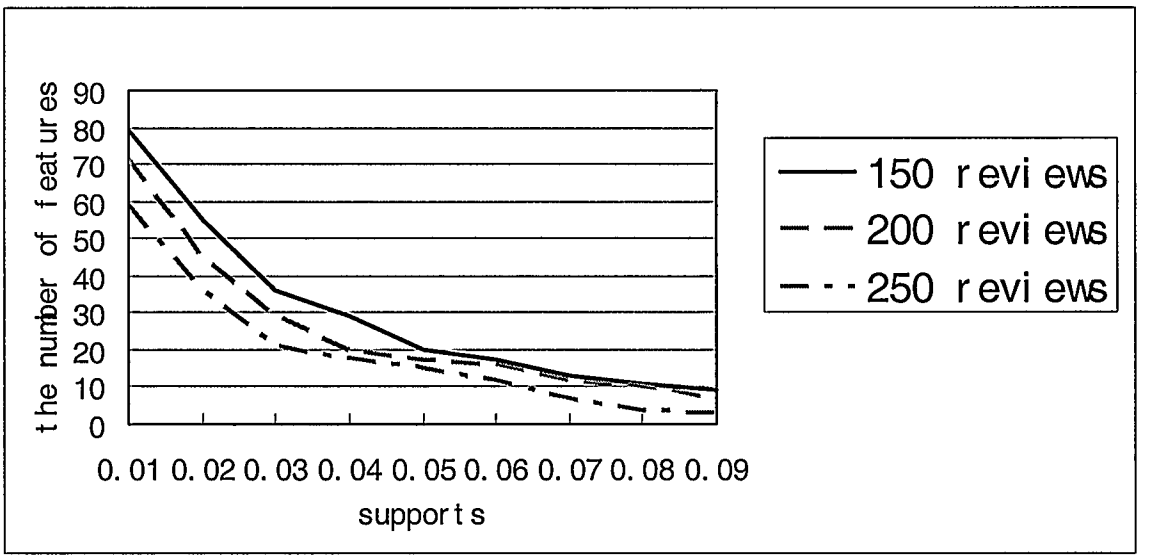


Figure 14: The relation between the number of features and the support parameters of the closure algorithm.

In Figure 14, the optimal 15 features are found at the supports of 0.065, 0.060, and 0.058 for 150 reviews, 200 reviews, and 250 reviews respectively. Thus, if the number of reviews is around 200, then we choose the 0.060 as the support. The most noise word is the name of the product. It is noise because some reviews start with the name of the product as introduction. This does not mean that the name of the product is one of the features.

To achieve the third goal, we calculate the values of minimum co-occurrence when the number of reviews is 150, 200, and 250. In Table 7, the first column lists the name of the noun phrases; the second, third, and fourth columns list the values of minimum co-

occurrence when the number of reviews is 150, 200, and 250 respectively. When analyzing the results, we found minimum co-occurrence of the noun phrase "stem capability" from the road bike reviews is much lower than other features'. After manually reading the road bike reviews again, we think the primary reason making the result lower is that most product reviewers of the road bike more like to use "stem" as a feature word instead of the full noun phrase "stem capability". If the number of reviews is around 200, then we choose 63% as minimum co-occurrence.

Table 7: The relation between the number of features and minimum co-occurrence

Name of noun phrases	Minimum co-occurrence		
	150 reviews	200 reviews	250 reviews
SD card (Digital Camera)	72%	73%	71%
SD card (PDA)	71%	70%	69%
Flash Memory (PDA)	78%	78%	77%
Stem capability (Road Bike)	32%	31%	30%
Average	63%	63%	62%

4.3 Evaluating the opinion extraction

We also experimented with how well the MOP system worked on extracting opinion orientations from the opinion sentence. For every opinion sentence, our system decides if the opinion sentence expresses a positive, negative, or neutral attitude.

Table 8: Recall and precision of opinion sentence extraction in MOP and in Hu and Liu's work.

Dataset	Opinion sentence extraction			
	MOP		Hu and Liu's work	
	Recall	Precision	Recall	Precision
Canon A570IS	0.84	0.81	0.73	0.69
Canon SD1000	0.86	0.84	0.72	0.81
Panasonic Lumix DMC-TZ3A	0.84	0.82	0.78	0.75
Nikon D40	0.82	0.80	0.77	0.71
Palm TX handheld PDA	0.80	0.79	0.75	0.72
PalmOne Zire 31 Handheld PDA	0.82	0.81	0.72	0.70
GMC Denali Road Bike	0.74	0.72	0.69	0.68
Average	0.81	0.80	0.74	0.72

Table 8 lists the precision and recall of the opinion sentence extraction generated by the MOP system. The names of product datasets are listed in Column 1. They are the same as the datasets we used for testing the feature extraction in Table 6. Column 2 and Column 3,

respectively, list the recall and precision of the opinion sentence extraction generated by the MOP system. The recall and precision of the opinion sentence extraction generated by the Hu and Liu's work are listed in Column 4 and Column 5.

In the MOP system, the average recall of opinion sentence extraction is 0.81 and the average precision is 0.8. Clearly, the results generated by the MOP system are still better than those generated by Hu and Liu's work. The primary reason is that we extend the opinion words from adjectives to adjectives, verbs, nouns, and adverbs. However, extracting opinion sentence in the MOP system still has some errors. When analyzing mining results, we found that the MOP system cannot incorporate pronoun resolution. For example, consider an opinion sentence, "So, I buy this camera because of the quality. It is really great", where the pronoun "it" replaces the "quality". The opinion sentence evaluates the camera's quality and indicates the opinion orientation is positive. However, the MOP system identifies that the opinion orientation of the feature "quality" is neutral. The reason is that the opinion sentence (the first sentence) containing the feature "quality' does not contain the opinion word "great" that is in the next sentence. Thus, it decreases the recall and precision of the opinion sentence extraction. We also found that some description sentences lower precision of opinion sentence extraction, where the description sentences include how owners get the products, how to use the products, how the products achieve the promise, *etc.* Sometimes, these sentences contain feature words and opinion words, but the product reviewers do not try to express any attitudes about the product or its features.

Table 9: Sentence orientation prediction.

Datasets	Opinion sentence extraction	
	MOP	Hu and Liu's work
Canon A570IS	0.93	0.90
Canon SD1000	0.92	0.92
Panasonic Lumix DMC-TZ3A	0.90	0.87
Nikon D40	0.94	0.81
Palm TX handheld PDA	0.91	0.86
PalmOne Zire 31 Handheld PDA	0.88	0.84
GMC Denali Road Bike	0.90	0.78
Average	0.91	0.85

Finally, Table 9 shows that our system also has a better accuracy in predicting opinion sentence extraction orientations than Hu and Liu's work. The main reason is that we use the KFNC method and extend the opinion words to adjectives, verbs, nouns, and adverbs, which improve the accuracy in predicting opinion sentence orientation. Additionally, the MOP system missed the cases like "This camera couldn't be better" because we take the negation of the word "better" due to the word "not". Our system indicated the opinion orientation of the opinion sentence is negative, even though it should be positive, thereby decreasing the precision of the feature extraction in the MOP system.

4.4 Comparison with other Holistic Approaches

Recall from Chapter 2 that this work is most closely related to that of Dang *et al.* [60] in that they take a holistic approach using a lexicon-based approach to find opinion words by exploiting external evidence and linguistic convention. Our work is similar to this in that it draws upon external information such as WordNet to help determine the opinions being expressed. Our work includes additional features beyond theirs such as:

- The user can input infrequent features of products into the MOP system to extract opinions, which is described in Section 3.2.3.
- Our system extends the opinion words to capture adjectives, verbs, adverbs, and nouns, which is described in Section 3.2.4.
- The MOP system lowers the bias from product reviewers arising due to specific knowledge and particular requirements by setting a vote rating weighted value for each product reviewer (as discussed in Section 3.2.7).
- The affinity is calculated by comparing the positive overall evaluation results of a product generated by the MOP system with its average product review value posted online to determine if the average product review can be trustworthy, which is described in Section 3.2.8.

Deng *et al.* [60] empirically demonstrates their system by acquiring some stunning results. For eight consumer products, the average recall of opinion sentence extraction is 91% and average precision is 90%. These results are excellent and only rarely seen in the literature and our results have not been able to duplicate them as indicated in this chapter. However, the results are based on a relatively small set of products and the influence of "external evidence" is somewhat ambiguous in their work so it is difficult to assess how

much "guidance" was provided to their miner. We have not been able to reproduce these results but we acknowledge they are impressive.

However, we argue that our approach minimizes the amount of external evidence required and provides linguistic support that is readily available from public sources. We would also argue that our approach is generic in that it can be applied to wide range of different products without adding any additional external evidence or linguistic guidance. The components of our system, that are very similar to their approach, fail to produce results anywhere near the ones they report but we suspect that we are not providing as much guidance in our system on a per product basis. This will obviously result in a greater reliance on the automatic features of the system but it would be interesting to run an experiment with our system that incorporates all of their external information and/or to run an experiment with their system where it only used the amount of external information that we draw upon in our implementation. This would provide an excellent and fair comparison, but neither of these options was available at the time of writing so we will have to leave it in the realm of future research.

Chapter Five: Conclusion

5.1 Summary of contributions

This thesis proposes an opinion mining system called MOP, which is a feature-based opinion mining system. The mining results generated by our system are the product's overall positive scores, negative scores and a list of its frequent and infrequent features with positive, negative or negative scores. It is described in Section 3.2.3.

In Chapter 2, we mainly describe the related work of two tasks involved in our MOP system, extracting features in Section 2.1 and identifying opinion orientations in Section 2.2. For extracting features, we discuss several related work, such as document retrieval, word co-occurrence extraction, a language model used to extract features, and feature extraction by part of speech. For identifying opinion orientations, we sequentially review the genre classification and some previous work related to opinion extraction from product reviews.

In Chapter 3, we first demonstrate the design of the MOP system in Section 3.1 and then describe the detail components of our system in Section 3.2. The MOP system is decomposed into eight steps. Corresponding to each step in the system design, the detail component description of our system consists of eight sections: crawling reviews, tagging part of speech, identifying frequent and infrequent features, extracting opinion words,

classifying opinion orientations, refining feature candidates, generating summary, and calculating affinity. In this chapter, we provide four thesis contributions as below:

1. URL is the only required input in the MOP system.
2. Opinion words have been extended to be adjectives, verbs, adverbs, and nouns.
3. A vote rating weighted value for each product reviewer has been set to lower the bias arising from product reviewers' specific knowledge and particular requirements.
4. The affinity is calculated by comparing the positive overall evaluation result of a product generated by the MOP system with its average product review posted online to determine if the average product review is trustworthy.

In Chapter 4, our empirical study evaluates the MOP system in three perspectives: first, the effectiveness of feature extraction in Section 4.1; second, the effectiveness of opinion sentence extraction in Section 4.2; third, the accuracy of orientation prediction of opinion sentences in Section 4.3; and forth, comparison of our system with other holistic approaches in Section 4.4.

5.2 Future work

There is rich potential for future work. We will further improve and refine our system from four aspects. First, we will incorporate pronoun solution to improve the feature extraction. For example, there is an opinion sentence, "I have to mention the battery life. It is very good." Here, the pronoun "it" could be replaced by the noun word "battery" from the previous sentence. Then, the opinion word "good" indicates that the attitude towards the "battery" is positive. Now the MOP system indicates the attitude towards the "battery" is neutral, which is wrong.

Second, some product reviewers like to write the symbol "pics" instead of the word "pictures" to save the words in their consumer product reviews. In our current work, we cannot recognize that the "pics" represents the word "pictures". In the future, we could leverage search engines to determine if the symbol "pics" is similar to the word "pictures".

Third, we will also incorporate some simple and often used language patterns in English language to correctly extract the opinion orientations of the sentences. For example, "I think this camera couldn't be better". Humans can easily determine the opinion orientation of the above sentence to be positive. However, the MOP system first detects that the opinion word "better" indicates a positive attitude, but there is an adverb "not" in the sentence. The MOP system indicates that the sentence is negative. Obviously, the result is incorrect.

Fourth, we will extend opinion words extraction to sequence sentences. For instance, there are two sequential opinion sentences, "Before using it, I thought this printer is good. Actually, not!". The first opinion sentence shows a positive opinion for the printer, but the second sentence suddenly negatives it. Unfortunately, the MOP system indicates "the printer" to be positive. However, it is wrong.

Fifth, we also like to extend our approach to process blog articles with clustering of blog topics.

References

[1] A. Andreevskaia and S. Bergler, "Mining WordNet for Fuzzy Sentiment: Sentiment Tag Extraction from WordNet Glosses," *Proceedings of EACL 2006.*

[2] N. J. Belkin and W. B. Croft, "Retrieval Techniques," *Annual Review of Information Science and Technology,* vol. 22, 1987.

[3] F. Benamara, C. Cesarano, A. Picariello, D. Reforgiato, and V. S. Subrahmanian, "Sentiment analysis: Adjectives and adverbs are better than adjectives alone," *Proceedings of ICWSM,* vol. 7, pp. 203-206, 2007.

[4] R. F. Bruce and J. M. Wiebe, "Recognizing subjectivity: a case study in manual tagging," *Natural Language Engineering,* vol. 5, pp. 187-205, 1999.

[5] C. Cardie, J. Wiebe, T. Wilson, and D. Litman, "Combining Low-Level and Summary Representations of Opinions for Multi-Perspective Question Answering," *Working Notes-New Directions in Question Answering (AAAI Spring Symposium Series),* 2003.

[6] R. Carter and M. McCarthy, *Cambridge Grammar of English: A Comprehensive Guide: Spoken and Written English Grammar and Usage*: Cambridge University Press, 2006.

[7] J. A. Chevalier and D. Mayzlin, "The effect of word of mouth on sales: Online book review," *Journal of Marketing Research,* pp. 345-354, 2006.

[8] J. G. Conrad and F. Schilder, "Opinion mining in legal blogs," *Proceedings of the 11th international conference on Artificial intelligence and law,* pp. 231-236, 2007.

[9] R. Cowan, *The Teacher's Grammar of English: A Course Book and Reference Guide*: Cambridge University Press, 2008.

[10] D. Cristofor, L. Cristofor, and D. A. Simovici, "Galois Connections and Data Mining," *Journal of Universal Computer Science,* vol. 6, pp. 60-73, 2000.

[11] S. R. Das and M. Y. Chen, "Yahoo! for Amazon: Sentiment Extraction from Small Talk on the Web," *Management Science,* vol. 53, p. 1375, 2007.

[12] K. Dave, S. Lawrence, and D. M. Pennock, "Mining the peanut gallery: opinion extraction and semantic classification of product reviews," *Proceedings of the 12th international conference on World Wide Web,* pp. 519-528, 2003.

[13] G. DeJong, "An overview of the FRUMP system," *Strategies for Natural Language Processing,* pp. 149-176, 1982.

[14] A. Esuli and F. Sebastiani, "Determining term subjectivity and term orientation for opinion mining," *Proceedings the EACL 2006,* 2006.

[15] J. L. Fagan, "The effectiveness of a nonsyntactic approach to automatic phrase indexing for document retrieval," *Journal of the American Society for Information Science,* vol. 40, pp. 115-132, 1989.

[16] C. Fellbaum and I. NetLibrary, *WordNet: an electronic lexical database*: MIT Press USA, 1998.

[17] A. Finn, N. Kushmerick, and B. Smyth, "Genre Classification and Domain Transfer for Information Filtering " in *European Colloquium on Information Retrieval Research*, 2002, pp. 353-362.

[18] J. Goldstein, M. Kantrowitz, V. Mittal, and J. Carbonell, "Summarizing text documents: sentence selection and evaluation metrics," *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval,* pp. 121-128, 1999.

[19] V. Hatzivassiloglou and K. R. McKeown, "Predicting the semantic orientation of adjectives," *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics,* pp. 174-181, 1997.

[20] V. Hatzivassiloglou and J. M. Wiebe, "Effects of adjective orientation and gradability on sentence subjectivity," *Proceedings of the 18th conference on Computational linguistics-Volume 1,* pp. 299-305, 2000.

[21] M. A. Hearst, "Direction-based text interpretation as an information access refinement," *Text-Based Intelligent Systems: Current Research and Practice in Information Extraction and Retrieval. Mahwah, NJ: Lawrence Erlbaum Associates,* 1992.

[22] J. Hipp, U. Gutzer, and G. Nakhaeizadeh, "Algorithms for association rule mining general survey and comparison," *ACM SIGKDD Explorations Newsletter,* vol. 2, pp. 58-64, 2000.

[23] H. Kanayama, T. Nasukawa, and W. Hideo, "Deeper sentiment analysis using machine translation technology," *Proceedings of the 20th international conference on Computational Linguistics,* 2004.

[24] M. Hu and B. Liu, "Mining Opinion Features in Customer Reviews," *Proceedings of the national conference on artificial intelligence,* pp. 755-760, 2004.

[25] M. Hu and B. Liu, "Mining and summarizing customer reviews," *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining,* pp. 168-177, 2004.

[26] N. Jindal and B. Liu, "Identifying comparative sentences in text documents," *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval,* pp. 244-251, 2006.

[27] H. Kanayama and T. Nasukawa, "Fully automatic lexicon expansion for domain-oriented sentiment analysis," *Proceedings of EMNLP 2006,* 2006.

[28] J. Karlgren and D. Cutting, "Recognizing text genres with simple metrics using discriminant analysis," *Proceedings of the 15th conference on Computational linguistics-Volume 2,* pp. 1071-1075, 1994.

[29] B. Kessler, G. Numberg, and H. Schuze, "Automatic detection of text genre," *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics,* pp. 32-38, 1997.

[30] S. M. Kim and E. Hovy, "Determining the sentiment of opinions," *Proceedings of COLING,* vol. 4, pp. 1367-1373, 2004.

[31] L. W. Ku, L. Y. Lee, T. H. Wu, and H. H. Chen, "Major topic detection and its application to opinion summarization," *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval,* pp. 627-628, 2005.

[32] B. Liu, R. Grossman, and Y. Zhai, "Mining data records in Web pages," *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining,* pp. 601-606, 2003.

[33] B. Liu, M. Hu, and J. Cheng, "Opinion observer: analyzing and comparing opinions on the Web," *Proceedings of the 14th international conference on World Wide Web,* pp. 342-351, 2005.

[34] Y. Matsuo and M. Ishizuka, "Keyword Extraction from a Single Document Using Word Co-Occurrence Statistical Information," *International journal on artificial intelligence tools,* vol. 13, pp. 157-170, 2004.

[35] Q. Mei, X. Ling, M. Wondra, H. Su, and C. X. Zhai, "Topic sentiment mixture: modeling facets and opinions in weblogs," *Proceedings of the 16th international conference on World Wide Web,* pp. 171-180, 2007.

[36] G. Mishne and M. de Rijke, "MoodViews: Tools for blog mood analysis," *AAAI 2006 Spring Symp. on Computational Approaches to Analysing Weblogs (AAAICAAW 2006),* 2006.

[37] S. Morinaga, K. Yamanishi, K. Tateishi, and T. Fukushima, "Mining product reputations on the Web," *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining,* pp. 341-349, 2002.

[38] R. Murphy, B. Viney, and M. Craven, *English grammar in use*: Cambridge University Press Cambridge, 1994.

[39] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10,* pp. 79-86, 2002.

[40] J. Park and D. Barbosa, "Adaptive record extraction from web pages," *Proceedings of the 16th international conference on World Wide Web,* pp. 1335-1336, 2007.

[41] S. Piao, S. Ananiadou, Y. Tsuruoka, Y. Sasaki, and J. McNaught, "Mining Opinion Polarity Relations of Citations," *International Workshop on Computational Semantics (IWCS),* pp. 366-371, 2007.

[42] A. M. Popescu and O. Etzioni, "Extracting product features and opinions from reviews," *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing,* pp. 339-346, 2005.

[43] R. Richardson, A. F. Smeaton, and J. Murphy, "Using WordNet as a Knowledge Base for Measuring Semantic Similarity between Words," *Proceedings of AICS Conference. Trinity College, Dublin,* 1994.

[44] R. Richardson and A. F. Smeaton, "Using WordNet in a Knowledge-Based Approach to Information Retrieval," *Proceedings of the BCS-IRSG Colloquium, Crewe,* 1995.

[45] W. Sack, "On the computation of point of view," *Proceedings of the twelfth national conference on Artificial intelligence (vol. 2) table of contents,* 1994.

[46] G. Salton, A. Singhal, C. Buckley, and M. Mitra, "Automatic text decomposition using text segments and text themes," *Proceedings of the the seventh ACM conference on Hypertext,* pp. 53-65, 1996.

[47] C. Scaffidi, K. Bierhoff, E. Chang, M. Felker, H. Ng, and C. Jin, "Red Opal: product-feature scoring from reviews", *Proceedings of the 8th ACM conference on Electronic commerce,* pp. 182-191, 2007.

[48] M. Sincich, *Statistics*, Ninth ed.: Prentice-Hall, 2003.

[49] P. Subasic and A. Huettner, "Affect analysis of text using fuzzy semantic typing," *Fuzzy Systems, IEEE Transactions on,* vol. 9, pp. 483-496, 2001.

[50] J. Tait, *Automatic Summarizing of English Texts* vol. Ph. D.: University of Cambridge, 1983.

[51] S. Teufel, A. Siddharthan, and D. Tidhar, "Automatic classification of citation function," *Proceedings of EMNLP,* vol. 6, pp. 103-110, 2006.

[52] R. Tong, "An operational system for detecting and tracking opinions in on-line discussion," in *SIGIR 2001 Workshop on Operational Text Classification*, 2001.

[53] P. D. Turney, "Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews," *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics,* pp. 417-424, 2002.

[54] J. Wiebe and R. Mihalcea, "Word Sense and Subjectivity," *Annual meeting-association for computational linguistics,* vol. 44, p. 1065, 2006.

[55] J. M. Wiebe, R. F. Brace, and T. P. O'Hara, "Development and Use of a Gold-Standard Data Set for Subjectivity Classifications," *Annual meeting-association for computation linguistics,* vol. 37, pp. 246-253, 1999.

[56] J. M. Wiebe, "Learning Subjective Adjectives from Corpora," *Proceeding of the national conference on artificail intelligence,* pp. 735-741, 2000.

[57] H. Yu and V. Hatzivassiloglou, "Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences," *Proceedings of EMNLP,* vol. 3, 2003.

[58] W. Zhang, C. Yu, and W. Meng, "Opinion retrieval from blogs," *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management,* pp. 831-840, 2007.

[59] L. Zhuang, F. Jing, and X. Y. Zhu, "Movie review mining and summarization," *Proceedings of the 15th ACM international conference on Information and knowledge management,* pp. 43-50, 2006.

[60] X. W. Deng, B. Liu, and P. S. Yu, " A holistic lexicon-based approach to opinion mining," *Proceedings of the international conference on Web search and web data mining,* pp. 231-240, 2008.