Web Opinion Mining on Consumer Reviews

WONG, Yuen Chau

A Thesis Submitted in Partial Fulfillment
of the Requirements for the Degree of
Master of Philosophy
in
Systems Engineering and Engineering Management

September 2008

Thesis/Assessment Committee

Professor LAM, Wai (Chair)

Professor YANG, Chuen-Chi, Christopher (Thesis Supervisor)

Professor YU, Xu, Jeffrey (Committee Member)

Professor Goh Dion (External Examiner)

ABSTRACT

The World Wide Web gathers useful information and users' opinions from all over the world. Searching and analyzing others' opinions from the web become a common practice for most users before making decision, but we always face the problem of information overloading – the information supply exceeds our actual needs. Based on the above facts, this work focuses on the extraction of sentences which describe various product features from consumers' review websites automatically. We employ concept clustering to organize terms that describe the same idea (also known as concept) into groups. Concept clusters which are related to product features are used to generate classifiers for the identification of product feature sentences. A new clustering algorithm named Scalable Distance Clustering Algorithm is proposed to improve the flexibility in clusters expansion. Experiments show that our proposed algorithm work more effectively in minimizing the number of non-product feature concept clusters while maximizing the accuracy of the terms in the product feature concept clusters compare with existing methods.

摘要

現今的互聯網結集了來自世界各地的資訊及用家意見。不少網絡用家都喜歡在決策前先搜尋和分析這類資料，但經常遇到資訊超載 (information overloading) —— 即用家取得的資訊往往比所需爲多 —— 的問題。有見及此，本論文將集中於如何自動有效地收集來自網上購物網站內用家填寫有關產品特點的意見字句。首先，我們會運用概念分群 (concept clustering) 把用於描繪同一概念的字詞分成相應的群組。有關產品特點的群組將會被擷取，並變成分類器用以找出相關的意見字句。本論文還提出了一個新的分群方法名爲可變距離分群法 (Scalable Distance Clustering Algorithm) 來改善群組擴張時的靈活性。從實驗所得，可變距離分群法比現有的方法能更有效地減少與產品特點無關的群組，並同時增加有關產品特點的群組中的字詞準確性。

ACKNOWLEDGEMENT

I would like to express my gratitude to all those who gave me support in completing this thesis. I would like to thank my supervisor, Professor Chris Yang, for giving me valuable advice and encouragement in these two years. With his enthusiasm and inspiration, he helped to make this thesis possible. I also wish to thank Professor Wai Lam and Professor Jeffrey Yu to be my thesis committee members and give me suggestion to improve this work.

My thanks and appreciation also goes to my fellow schoolmate, especially Dickson, Sampson, Eddie, Alex, Henry, Anthony and Sharon for helping me get through the difficult times, serving as my emotional support and entertainment. I am also grateful to my entire high school friends, particularly Kimmy, Olivia, Pui, Steve, Victor, Alan, Chiffon and Lam for the caring they provided.

Lastly, I would like to give my special thanks to my family for providing a comfortable environment for me. Their patient love and endless support enabled me to complete this work.

TABLE OF CONTENTS

1. Introduction

1.1 Overview

The World Wide Web undergoes a revolution in these few years. Web users are now changing from passive receivers to active contributors. In year 2006, "You" were even chosen to be the Time's Person of the Year due to "your" contribution to the Web. The magazine also concluded that "It's about the many wresting power from the few and helping one another for nothing and how that will not only change the world, but also change the way the world changes."[1]

Nowadays, the web touches all aspect of our lives. We can shop, broadcast video, buy movie tickets, read news, etc, through various websites. No matter what services a web site provide, most of them share a common feature – allowing web users to give comments. This open platform plays an important role in facilitating the spread of information. Imagine you would like to buy a digital camera and want to compare several similar models before purchase. In the past, you can only get the related information from leaflet, magazines or sales person, which may not suit your actual needs and even have bias. Now, with the help of some online shopping sites, like Amazon.com, consumers can give rating and comments on the products being sold through the sites. Consumers can read through those reviews from various websites to compare the strengths and weaknesses of different products, so that they can make a wise choice based on others' experience. These feedbacks are also helpful for the manufacturers as they can have a better understanding to the needs of different customers as well as the strength and weakness of their competitors' and their own products, so that they can improve their products and marketing strategies. Other than the online shopping sites, web sites that contain news, movies' information, restaurants searching function, etc, like yahoo.com, also welcome users to score and comment on the corresponding topics. Such comment is a valuable source of information for the general users to make comparison across different products and understand others' point of view. At the same time, companies or

[1] "Time's Person of the Year: You" *Time*, December 13, 2006.

producers can figure out the strength and weakness of their competition and themselves and take corresponding action.

Besides the web sites being discussed, web users can also join different discussion groups (also known as forums) to discuss their interested topics. The difference between typical web pages and forums is that the discussion topics in a forum are initiated by the web users themselves. The topics being discussed can be ranging from serious issues (e.g. social problems, government policies, news article) to informational (e.g. soccer matches, cooking receipts, travelling tips) or even for entertainment (e.g. online games, funny stories). The advantage of using discussion group is that it can easily gather people who share similar interests from all over the world. Web users can even build a personalized weblog with the help of some well-known web sites like Blogger.com. A blog is actually a web site, where you can write you own stuff and connect with others by joining a blog ring. Users in the same blog ring may share some similarity, for example, they may graduate from the same high school or they are all NBA's fans.

With the help of those websites, discussion groups, forums and the large scale communities, the web has become a valuable source for gathering different information and users' opinions. However, the downside is that the amount of information being available greatly exceeds the users' needs. For some hot topics, the number of opinions can increase exponentially. It is always time consuming if we analyze all the opinions by human effort. Moreover, even within the same topics, it can be further divided into a number of sub-categories. Individuals may be interested in some specific sub-topics only, for example, one may want to compare the battery life of the digital camera but not all the information about the digital camera. In addition, some information may not be so well-structured, i.e. related topics may locate in different sections or pages, especially those obtained in the discussion group and blog, extra effort is needed to obtain the desired information. In order to extract the useful information with minimized human effort, different techniques are now being developed to identify the possible features (sub-categories) for each single topic or product automatically or semi-automatically, such that tailor-made information can be obtained for each individual.

1.2 Motivation

The retrieval of useful information from web users' comments and consumers' reviews can be subdivided into a sequence of tasks. Since a review may comment on more than one product features, it is better to analysis each sentences rather than the review as a whole. For each sentence, we can determine whether it is related to any product features. If the sentence comment on a product features, we can further specify its orientation (positive, negative or neutral). A feature-based summary can be produced with the extracted information.

This sequence of tasks can fall into the categories of sentiment analysis and sentiment classification in the research area. Sentiment analysis refers to the extraction of product features and their corresponding polarity while sentiment classification focuses on the identification of the polarity only. Most of them employ the natural language processing technique to locate the opinion expression (Liu et.al 2005, Yi et. al 2003, Popescu & Etzioni 2005, Zhang et. al 2006, Dave et. al 2003, Turney 2001, Hu & Liu 2004, Scaffidi et. al 2007, Jindal & Liu 2008, Yu & Hatzivassiloglou 2003 and Ding et. al 2008). Since NLP approach relies on the part-of-speech and grammar rules to generate language patterns, the accuracy can be greatly influenced by the writing quality. As online opinions are usually written informally without spelling or grammatical check, loss of useful information can be huge by using the NLP approach. According to Liu et.al (2005), only 52% of the data can be correctly tagged, showing that there is still room to improve.

The tasks also relate to document classification, which helps to assign documents with similar contents into groups or extract the documents that are closed to a user-input query. Documents are transformed to a vector representation format after the removal of general terms (e.g. a, an, the...). Most similarity measures between two documents depend on the co-occurrence of the terms in the vectors. Hence, if sufficient information is provided, i.e. with sufficient number of documents for each class and with sufficient number of terms in each document, the accuracy of classifiers or the quality of clusters can be improved. Unlike document, the number

of terms in a sentence is limited. Also, different words can be used to describe the same product feature, but they seldom appear in the same sentence. Consider the following two sentences, "The LCD is bright and glossy." and "This camera has a high quality screen.", both of them describe the same product feature "LCD screen", but they don't have any co-occurred terms. Therefore, we cannot apply the techniques used in document classification directly in solving this problem.

Typical text mining techniques are also helpful in retrieving useful pattern from the reviews and in fact sentiment analysis, sentiment classification and document classification are all developed based on the idea of text mining. In typical text mining or data mining, a sentence can also be transformed to a bag of words while the general terms are removed. For supervised learning methods, such as Association Rules Mining, Naïve Bayesian classifier and Support Vector Machine, labeled training set is needed and the sentences can only be labeled with the predefined classes. Since the features being commented change from time to time, it is impossible to prepare a labeled training set for every topic. As a result, the supervised learning method is not suitable to analysis text with fast changing content. For unsupervised learning method, such as K-means, Hierarchical Clustering, Density-Based Scan, it suffers from the same problem as document classification, i.e. lack of co-occurrence term.

Sentence-based analysis suffers from the problem of noisy data, since some web users would like to tell his/her story rather than the summary of his/her opinion in the consumer review sites. When we examine the reviews sentence-by-sentence, it is easy to observe that not all the sentences are associated with product features. Based on our collected data, less than 50% of sentences are truly describing the product features. That is one of the reason why document-based analysis is not applicable in solving the sentence-based problem since a document always has its own theme. As the existing algorithms cannot best fit to the current situation, we would like to tackle the problem with an alternative approach, which can reduce the noisy data and extract as many useful clusters as possible.

1.3 Objective

In this work, we address the problem of extracting product feature sentences from consumer review websites. Let $R_p = \{r_{p\_1}, r_{p\_2}, \ldots, r_{p\_n}\}$ be a set of opinions (also known as reviews) of a particular type of product p (which can come from different brands or different models). For each review r_i, it is made up of a sequence of sentences, $S_i = < s_{i\_1}, s_{i\_2}, \ldots, s_{i\_m} >$. Let $T = \{ t_{ij\_1}, t_{ij\_2}, \ldots, t_{ij\_h} \}$ be a set of words, such that $s \subseteq T$.

A product feature f is a component or characteristic of a product that has been commented on some of the reviews. Let $F_p = \{f_{p\_1}, f_{p\_2}, \ldots, f_{p\_n}\}$ be a set of product features of a particular type of product p. Each product feature associate with a number of terms, i.e. $\{t_{p1\_1}, t_{p1\_2}, \ldots, t_{p1\_n}\} \rightarrow f_{p1}$, such that $t \subseteq T$. The terms should be able to identify the corresponding product feature sentences.

By comparing the terms appear in the sentences and the associated terms of each product feature, we can group the sentences which have commented on the same product feature. Let $C_p = \{c_{f\_1}, c_{f\_2}, \ldots, c_{f\_n}\}$ be the set of clusters subdivided from the review, R_p, For each cluster, it is made up of a group of sentences related to the same product feature f, such that $c_f = \{ s_{f\_1}, s_{f\_2}, \ldots, s_{f\_m} \}$ with $s \in S$ and $f \in F$.

1.4 Our contribution

We first investigated the strengths and weaknesses of various well-known supervised and unsupervised learning techniques for extracting product features sentences. We then studied the effectiveness of concept clustering in tackling this problem. A concept refers to an abstract idea that can denote all the entities of a topic and the set of entities is called concept cluster. For example, "battery life" and "charging time" can be used to describe "battery" while "AAA" and "lithium cell" are different battery types. A concept cluster about "battery" can be formed by the four terms.

We also proposed a new clustering algorithm named "Scalable Distance Clustering Algorithm" to extract concept clusters. By using our proposed algorithm, only clusters with strong internal association power between the entities can be found, i.e. all the terms being used to describe a concept are having strong association. We believe that the core of a cluster can better represent a concept and the strength decrease near the border, hence during the formation of a new concept cluster, such information is also recorded as a reference. By comparing the concept cluster with the extracted sentences, sentences which comment on the same concept are grouped together.

The Scalable Distance Clustering Algorithm can reduce the noisy data successfully and at the same time it can capture the useful entities efficiently. Experiments show that it out-performs the existing clustering algorithms, DB Scan and Hierarchical clustering, in solving this problem. The details of the algorithm and the experiment are shown in later sections.

1.5 Organization of the Thesis

Chapter 2 introduces some related researches about the extraction of product feature sentences. In chapter 3, we employ some supervised learning algorithms to solve the problem and discuss the limitation of supervised learning. In chapter 4, we use unsupervised learning algorithm to cluster the sentences. Chapter 5 discusses the idea of concept clustering and we will try to apply this technique to extract product feature keyword clusters and then use the keywords to generate rules for the classification of product feature sentences. In chapter 6, we further propose a clustering algorithm named Scalable Distance Clustering Algorithm for concept clustering. Finally, we give a conclusion and discuss some possible future works in the last chapter.

2. Related Work

Opinion mining is a sub-topic of text mining. It is related to sentiment classification and sentiment analysis. Most of the previous researches (Liu et.al 2005, Yi et. al 2003, Popescu & Etzioni 2005, Zhang et. al 2006, Dave et. al 2003, Turney 2001, Hu & Liu 2004, Scaffidi et. al 2007, Jindal & Liu 2008, Yu & Hatzivassiloglou 2003 and Ding et. al 2008) apply the linguistic rules or nature language processing (NLP) techniques to identify the product feature and orientation of online documents or sentences.

2.1 Existing Sentiment Classification Approach

Sentiment classification aims to identify the polarity for the given set of documents. Hatzivassiloglou and McKeown (1997) use a log-linear regression model and linguistic constraints to predict the relationship between conjoined adjectives and then apply a clustering algorithm to distinguish the adjectives into positive and negative. Turney (2001) applies the natural language processing technique to identify phrases containing adjective or adverb in the consumer reviews. The semantic orientation of a phrase is obtained by comparing the pointwise mutual information between the given phrase and the term "excellent" with that of the phrase and the term "poor".

Yu and Hatzivassiloglou (2003) separate facts from opinion at document and sentence level using Naïve Bayesian Classifier. They then calculate the modified log-likelihood ratio for the set of words with their part-of-speech and a set of seed words. The cutoff parameters for identifying the orientation (positive or negative) of words are estimated based on training data and Monte Carlo analysis. Pang and Lee (2004) remove the objective sentences based on minimum cuts framework and then apply machine learning classifier to identify the subjective movie reviews into thumbs up or down.

In some cases, it is difficult to identify the polarity of a review, for example, it may be neutral for its orientation or the orientation can not be clearly defined. Therefore, it is better to note the polarity with flexible scale. Dave et. al (2003) use information retrieval technique to extract meaningful language patterns and assign score ranging from -1 to 1 to denote the polarity of patterns. These patterns are then used to separate positive and negative reviews with sentences as the basic unit. In 2005, Pang and Lee further rate the movie reviews with flexible scales from 0 to 3, instead of thump up or down only. They also apply a meta-algorithm based on metric labeling to ensure similar terms receive similar rating.

Other than applying NLP technique, alternative approaches and different domain have also been studied. Ku et. al. (2006) focus on capturing the orientation of Chinese news and blog post based on dictionaries. The extraction algorithm is built with a bottom-up style from words level to sentences level and finally view the document as a whole. Choi et. al. (2006) extract the opinion-related entities and relations using conditional random fields. They use binary integer linear programming approach to retrieve the relations between opinion expression and source entities in newswire articles.

Some researchers investigate the strengths and weaknesses of different techniques. Cui et. al. (2006) compare the effectiveness of different classifiers, including passive-aggressive algorithm based classifier, language modeling based classifier and winnow classifier. They also study the impact of higher order n-gram using large scale data set which is approximately 100K. Pang et. al. (2002) examine the effectiveness of applying machine learning techniques (Naïve Bayesian Classifier, maximum entropy classification and support vector machine) to the sentiment classification problem.

2.2 Existing Sentiment Analysis Approach

Sentiment analysis focuses on the extraction of sentiment features in the reviews and the polarity of the corresponding features. Most of the existing works develop their models based on natural language processing technique. Hu and Liu (2004) apply machine learning algorithm and word positioning technique to identify the feature terms and opinion terms. Nouns are used to identify product features while adjectives are used to indicate the opinion orientation. Yi et. al. (2003) compare the mixture language model and likelihood ration in the selection of features. They also propose Sentiment Analyzer to identify the association between the feature terms and the sentiment terms based on the sentiment pattern for a given sentence. Popescu and Etzioni (2005) introduce an unsupervised system OPINE which identify the explicit product features and rank opinions based on the strength of the sentiment orientation of the opinions. Ding et. al (2008) propose a holistic lexicon-based approach to handle opinion words that are domain dependent and with multiple conflicts based on their linguistic patterns.

Researchers also set up prototype systems which help the general web-user to compare the product features as well as the corresponding orientation. Liu et. al. (2005) apply supervised learning technique to generate language patterns to identify product features from Pros and Cons of reviews. A prototype system named as Opinion Observer which allows a visual comparison of multiple products according to different product features is implemented. Scaffidi et. al. (2007) base on the assumption that product feature are mentioned more frequently to generate a language model to extract the feature and base on the numeric score given by the web users to compute a new score for each feature. A user interface named Red Opal, which displays a summary of product feature, score and confidence, is also proposed.

Some existing works search for the relations between the consumer reviews and economic issues of the consumer products. Archak et. al (2007) analyze the relationship between online consumer reviews and the product demand. The

consumer review is first modeled as a combination of product feature and evaluation space. They then associate the transformed consumer reviews with product demand as a linear function. Ghose and Ipeirotis (2007) study the usefulness and impact of consumer reviews based on users' needs. The reviews are ranked according to the expected helpfulness and expected effect on sales for general consumer and manufacturer respectively.

Other works exploit the sentiment analysis technique with different applications. Liu et. al. (2006) identify the feature-product dependencies across multi-product consumer reviews by using decision tree classifier. They also implement a system which search for related comparative sentences based on user input queries. Jindal and Liu (2006) identify comparative sentences using machine learning techniques and the filtered comparative sentences are grouped into different categories according to the degree of comparative words. Jindal and Liu (2008) investigate the problem of opinion spam in consumer reviews and detect the spam using occurrence frequency and supervised learning technique.

Although most existing works focus on the analysis of consumer reviews, some of them employ similar techniques in other domain such as movie reviews and blog posts. Zhuang et. al. (2006) study the language pattern of movie reviews. They generate a list of feature-opinion pair by using the grammatical relation in the training data. Mishne (2005) classifies the mood of the web users according their blog posts. Support vector machine is used to extract a set of keywords for each mood. Mei et. al. (2007) propose Topic-Sentiment Mixture model to extract the topic (feature) and sentiment (polarity) of weblog articles based on probabilistic theories.

2.3 Our Approach

In this work, we aim to analyze online consumer product reviews. Since most online shopping sites, e.g. Amazon.com, allow users to score the overall performance of their products, this score provides a useful indicator to the polarity of the users' reviews. Moreover, a user may be interested in the details of the whole comments instead of the overall polarity only. Therefore, our work focus on the extraction of product feature sentences only.

Despite the shortcoming of NLP approaches as stated in previous sections, we use alternative methods for extracting the product feature sentences. The consumer reviews are treated as a bag of words without considering their part-of-speech and grammatical relations. Stop words and non-informative terms are removed by using various statistical measurements. This approach can help to minimize the effect of incorrect labeling due to grammatical mistakes and the use of newly-created internet language.

Our studies can be divided into three main parts. First, we employed the supervised learning methods to generate keyword-based models and discussed the limitation of supervised learning methods. Second, we moved on to categorize the sentences based on the product features they have commented on using unsupervised learning method. Third, we applied the concept clustering technique to generate concept-based models using existing unsupervised learning algorithm. Finally, we proposed a new clustering method named as Scalable Distance Clustering Algorithm to solve the concept clustering problem. Details are going to be discussed in the following chapters.

3. Extracting Product Feature Sentences using Supervised Learning Algorithms

3.1 Overview

Supervised Learning is a machine learning technique for modeling a classifier based on the given training datasets. The training data consists of pairs of data records and target class labels. Different supervised learning algorithms use different mechanism to analyze the training data to produce a unique classifier for predicting the class label of new coming data. A classifier can be represented in the form of association rules, decision tress, or mathematical modesl.

ID	Data Records	Class Labels
1	(0, 0, 0, 0)	X
2	(0, 0, 0, 1)	Y
3	(0, 0, 1, 0)	Z
4	(0, 0, 1, 1)	X
5	(0, 1, 0, 0)	Y
6	(0, 1, 0, 1)	Z
7	(0, 1, 1, 0)	X
8	(0, 1, 1, 1)	Y
9	(1, 1, 0, 0)	Z
10	(1, 1, 0, 1)	X

Figure 3.1 Sample Training Data

Figure 3.1 shows a sample training data with ten pairs of instances. Each instance consists of 4 binary numbers and a class label. By analyzing the relation between the 4-bit data records and class label with different supervised learning algorithms, different classifiers can be built. The classifiers can then be used to predict the class label when new data comes, e.g. (1, 1, 1, 1).

In this chapter, we explored the feasibility of the class association rule mining and naïve Bayesian classifier in the labeling of product features sentences. We also discussed the advantages and shortcomings of utilizing the existing supervised algorithms in sentiment analysis problem.

3.2 Association Rules Mining

Association rule mining was proposed by Agrawal et al in 1993 (Agrawal et. al 1993). It searches for relationships among items that satisfy the predefined minimum support and confidence within a given dataset. Let $S = \{s_1, s_2, ..., s_n\}$ be the set of transaction within a given dataset and $T = \{t_1, t_2, ..., t_n\}$ be the set of items, such that $s \subseteq T$. An association rule is actually an implication of the form:

$$A \rightarrow B, \text{ where } A, B \subset T \text{ and } A \cap B = \varnothing \quad \ldots \quad (3.1)$$

The support and confidence of $A \rightarrow B$ is given as follows:

$$support = \frac{(A \cup B).count}{n} \quad \ldots \quad (3.2)$$

$$confidence = \frac{(A \cup B).count}{A.count} \quad \ldots \quad (3.3)$$

3.2.1 Apriori Algorithm

Apriori Algorithm (Agrawal & Srikant, 1994) is a classic algorithm for mining frequent itemsets for association rules based on the Apriori property which states that all subsets of a frequent itemset must also be frequent. The generation of k-frequent itemsets, L_k, is done with two main steps. First, it joins the *(k-1)*-frequent itemsets, L_{k-1}, to form a candidate itemset, C_k. Then, it removes any *(k-1)*-itemsets that is not frequent from the candidate itemset, C_k. Finally, all association rules can be generated from the frequent itemsets with their confidence exceeds the predefined threshold.

3.2.2 Class Association Rules Mining

Class association rule mining is similar to normal association rule mining. The only difference is that normal association rule mining does not have any target item. But, for the class association rule mining, the target is specified by the user (also known as classes). Let $C = \{c_1, c_2, ..., c_n\}$ be the set of all classes that the user is interested in and $S' = \{< s_1, c_{s1} >, < s_2, c_{s2} >, ..., < s_n, c_{sn} >\}$ be another set of transaction. The formula of class association rules becomes as follows.

$$A \rightarrow c \text{, where } A \subset I \text{ and } c \in C \qquad \ldots \qquad (3.4)$$

3.3 Naïve Bayesian Classifier

3.3.1 Basic Idea

Naive Bayesian classifier is a probabilistic classifier based on Bayes theorem, which has a strong class conditional independence assumption. Let $C = \{c_1, c_2, ..., c_n\}$ be the set of all classes that the user is interested in. Given a data member $T = \{t_1, t_2, ..., t_n\}$, the classifier predict that T belongs to the class having the highest conditional probability

$$P(c_i \mid T) > P(c_j \mid T) \quad for\ all\ \ j \neq i \qquad \ldots \qquad (3.5)$$

The class c_i for which such that $P(c_i \mid T)$ is maximized is called a maximum posterior hypothesis. By Bayes Theorem,

$$P(c_i \mid T) = \frac{P(T \mid c_i) P(c_i)}{P(T)} \qquad \ldots \qquad (3.6)$$

Since $P(T)$ is constant for all classes, we only need to consider $P(T \mid c_i) P(c_i)$. In order to reduce the computation, the class conditionally independence assumption is

made. The equation is simplified and become equation (3.7). The prior probabilities $P(c_i)$ and the conditional probabilities $P(t_k \mid c_i)$ can be estimated from the training data.

$$P(c_i \mid T) = P(c_i) \prod_{k=1}^{m} P(t_k \mid c_i) \quad \ldots \quad (3.7)$$

3.3.2 Feature Selection Techniques

In a document, there are many irrelevant and redundant terms. These terms can greatly influence the processing speed and accuracy of the naïve Bayesian classifier. By removing these terms, we can also improve the interpretability and generalization capability of the classifier. It also help us to acquire better understanding about the data by telling which items are more important and how they are related with each other. In this section, different feature selection techniques are introduced. A combination of such techniques can be used in order to get a better result.

3.3.2.1 Term Frequency Thresholding

Term frequency is the number of sentences in which a term appears in the collection of sentences (Yang & Pedersen, 1997). Let $S = \{s_1, s_2, ..., s_n\}$ be the collection of sentences, the term frequency of a term t is given as followed.

$$TF(t) = |\{s_i : t \in s_i\}| \quad \ldots \quad (3.8)$$

Term frequency thresholding involves the removal of terms from the keywords list whose term frequency is less than a predefined threshold. The basic idea of term frequency thresholding is that rare terms are usually non-informative and do not have great influence to the overall performance of the classifier.

3.3.3.2 Information Gain

Information Gain measures the amount of information obtained for class prediction using the absence and presence of a term t (Yang and Pedersen, 1997). The term with high information gain can minimize the information needed to classify the datasets. The expected information needed to classify the given datasets by using certain attribute is called entropy. Let S be the whole set of data and $C = \{c_1, c_2, ..., c_n\}$ be the set of classes. To obtain the information gain of a term, we first use equation (3.9) to compute the entropy of the given datasets.

$$E(S) = -\sum_{i=1}^{k} P(c_i) \log_2 P(c_i) \quad \ldots \quad (3.9)$$

The prior probability $P(c_i)$ acts as a weighting factor and the log function to base 2 is used as there are two possible outcome, i.e. the presence or absent of a class. In compute the entropy of a term, we need to consider the distribution of the classes for the presence and the absence of that term t, the formula is given below.

$$E(t) = -P(t)\sum_{i=1}^{k} P(c_i \mid t) \log_2 P(c_i \mid t) - P(\bar{t})\sum_{i=1}^{k} P(c_i \mid \bar{t}) \log_2 P(c_i \mid \bar{t}) \ldots (3.10)$$

The information gain of a term is the expected reduction in entropy by considering the presence of the term, i.e. $G(t) = E(S) - E(t)$. The detail of the calculation is given in equation (3.11).

$$\begin{aligned} G(t) = & -\sum_{i=1}^{k} P(c_i) \log_2 P(c_i) \\ & + P(t)\sum_{i=1}^{k} P(c_i \mid t) \log_2 P(c_i \mid t) \\ & + P(\bar{t})\sum_{i=1}^{k} P(c_i \mid \bar{t}) \log_2 P(c_i \mid \bar{t}) \end{aligned} \quad \ldots \quad (3.11)$$

3.3.3.3 Chi Square Test

Chi Square evaluates statistically significant differences between proportions for a term and a class (Yang & Pedersen, 1997). It measures whether an observation on two variables, expressed in a contingency table, are independent of each other. The chi-square value between a term t and a class c is calculated as follows.

$$\chi^2(t,c) = \frac{[P(t \wedge c)P(\bar{t} \wedge \bar{c}) - P(t \wedge \bar{c})P(\bar{t} \wedge c)]^2}{P(c)P(\bar{c})P(t)P(\bar{t})} \quad \ldots \quad (3.12)$$

The goodness of a term is evaluated by comparing cross-categories chi square. Terms whose difference between the maximum chi square and the average chi square is less than a predefined threshold δ is removed.

$$\chi^2_{\max}(t) - \chi^2_{avg}(t) < \delta \quad \ldots \quad (3.13)$$

$$\chi^2_{\max}(t) = \max\{\chi^2(t,c_i)\} \quad \ldots \quad (3.14)$$

$$\chi^2_{avg}(t) = \sum_{i=1}^{k} P(c_i)\chi^2(t,c_i) \quad \ldots \quad (3.15)$$

3.4 Experiment

We collected a set of consumer reviews about digital camera and labeled the extracted sentences with user-defined product features. The set of sentences was partitioned into different combination of training and testing datasets. Supervised learning algorithms described in sections 3.2 and 3.3 were employed to generate different classifiers for the labeling of product feature sentences

3.4.1 Data Sets

We collected consumer reviews of digital camera from amazon.com randomly. The sets of reviews were decomposed into a sequence of sentences. Then they were further preprocessed by removing stop words and stemming. The distribution of the reviews is given in the following table.

Digital Camera Model	No. of sentence tagged	No. of reviews
Sony Cybershot DSCT7 5.1MP Digital Camera with 3x Optical Zoom (Includes Docking Station)	295	21
Canon EOS 20D 8.2MP Digital SLR Camera (Body Only)	564	34
Nikon D70 Digital SLR Camera Kit (Lens Included)	451	32
Canon Powershot SD300 4MP Digital Elph Camera with 3x Optical Zoom	352	25
Sony Cybershot DSCP200 7.2MP Digital Camera 3x Optical Zoom	338	28
Canon Powershot S2 IS 5MP Digital Camera with 12x Optical Image Stabilized Zoom	365	22
Canon PowerShot A95 5MP Digital Camera with 3x Optical Zoom	635	51
Total	**3000**	**213**

Table 3.2 Distribution of digital camera reviews

The sentences were all labeled manually with two main processes. We first generated a set of product features and their associated rules. The rules were separated by a semi-colon *";"*. In a single rule, the symbol *"{a / b}"* means either *"a"* or *"b"* appear while *"a + b"* means *"a"* and *"b"* co-exist in a sentence. The sentences are roughly tagged by some predefined rules. A sentence can be labeled with more than one product features or "NA" if it is not related to any one of them.

Product Features	Rules	# of sentences
battery	battery ; charge ; charger	149
flash	flash	83
image	color ; noise ; {picture / image / photo } + { quality / indoor / outdoor }	273
lens	lens ; zoom	241
memory	memory ; card ; mb ; gb	95
price	$; cost ; price ; money ; cheap ; expensive ; cheaper ; pay	156
screen	screen ; lcd	102
usability	menu ; mode ; setting ; easy / ease } + use ; manual ; control	333
video	video ; movie ; sound	67

Table 3.3 Human generated keyword lists for digital camera review

In the second steps, we examined all the sentences one by one and revise their class labels. If a product feature was not yet marked in the first step, that product feature and its associated rules are added to the list. We also kept track the performance of the rules and made amendment. The processes repeated until there was no change to the set of product features and all the sentences' labels. The finalized list of product features and the associated rules is listed in table 3.3.

3.4.2 Experimental Setup and Evaluation Measures

For supervised learning methods, we used 5-fold cross validation to estimate the overall performance of the algorithms. The original data, R_{dc}, was partitioned into 5 subsets. For each fold, a single subset was retained as the testing set while the remaining 4 subsets were combined to form the training set. The final result was estimated based on the average result from the folds. Precision, recall and f-score were used as the evaluation measures. For any product feature *f*,

$$r_f = \frac{TP_f}{TP_f + FN_f} \quad \ldots \quad (3.16)$$

$$p_f = \frac{TP_f}{TP_f + FP_f} \quad \ldots \quad (3.17)$$

$$f - score = \frac{2 \times p_f \times r_f}{p_f + r_f} \quad \ldots \quad (3.18)$$

, where

TP_f = number of sentences with feature f that are identified by both expert and machine.

FN_f = number of sentences with feature f that are identified by machine but not expert.

FP_f = number of sentences with feature f that are identified expert but not machine.

The evaluation measures can be further divided into micro point-of-view and macro point-of-view.

$$micro - r = \frac{\sum_{f \in F} TP_f}{\sum_{f \in F} TP_f + \sum_{f \in F} FN_f} \quad \ldots \quad (3.19)$$

$$micro - p = \frac{\sum_{f \in F} TP_f}{\sum_{f \in F} TP_f + \sum_{f \in F} FP_f} \quad \ldots \quad (3.20)$$

$$macro - r = \frac{\sum_{f \in F} r_f}{|F|} \quad \ldots \quad (3.21)$$

$$macro - p = \frac{\sum_{f \in F} p_f}{|F|} \quad \ldots \quad (3.22)$$

3.4.1 Class Association Rules Mining

Figure 3.1 and figure 3.2 show the change of macro f-score and micro f-score with setting different combination of support (0.005 – 0.007) and confidence (0.35 – 0.65) in class association rules mining respectively.

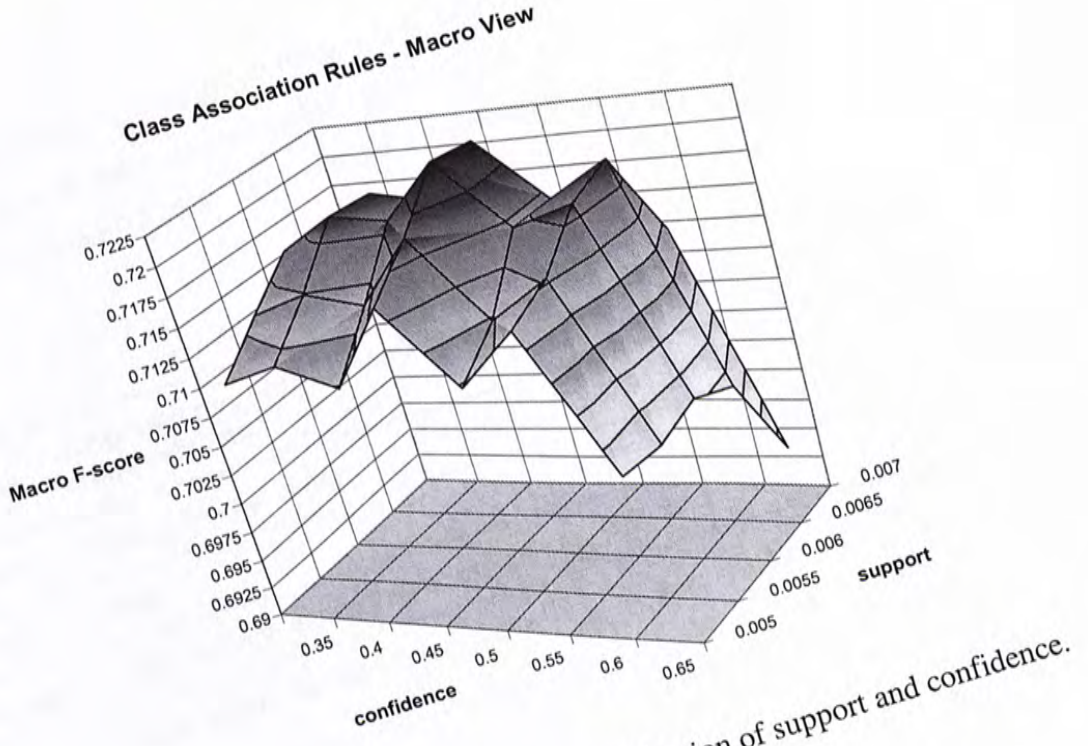

Figure 3.1 Macro F-score with different combination of support and confidence.

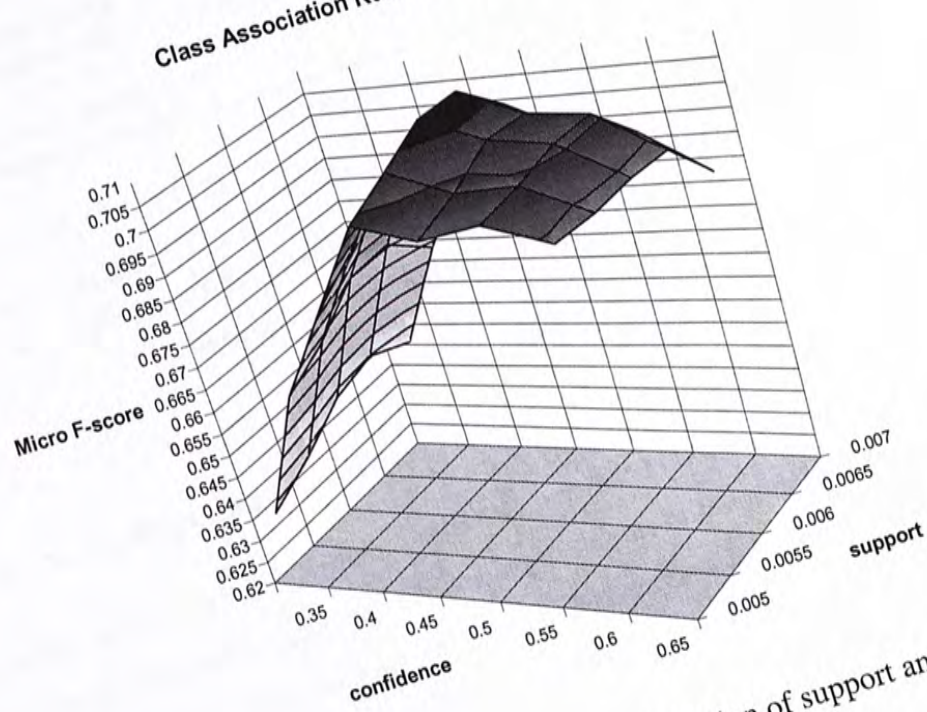

Figure 3.2 Micro F-score with different combination of support and confidence.

According to the above figures, the combination of support 0.006 and confidence 0.5 gave the best result to the class association rules methods with micro f-score 70.77% and best macro f-score 72.26%. The keyword list generated is listed is the following table.

Features	Integrated Keyword List in 5-fold cross validation	Recall	Precision	F-score
battery	*battery, life, aa, charger*	92.24%	82.29%	0.8698
flash	*flash*	89.76%	59.57%	0.7161
image	*color, quality, blurry, indoor, image*	45.03%	48.80%	0.4684
lens	*lens, zoom, optical, kit, telephoto, mm*	91.78%	67.04%	0.7748
memory	*memory, card, mb*	90.67%	67.56%	0.7743
price	*price, money*	52.79%	80.14%	0.6365
screen	*screen, lcd*	95.38%	74.80%	0.8385
usability	*menu, mode, setting, manual, control, function, easy, auto, button*	69.47%	63.71%	0.6646
video	*video, movie, sound*	70.83%	64.84%	0.6770
	Micro View	74.40%	67.47%	0.7077
	Macro View	77.55%	67.64%	0.7226

Table 3.4 Keyword list generated by class association rule mining by setting support = 0.006 and confidence = 0.5

3.4.2 Naïve Bayesian Classifier

As stated in section 3.3.2, keyword selection is needed in order to obtain a fair result. Since both Information gain and Chi square have bias in favor of rare keywords, and such information may mislead the classification, rare keywords (with sentence frequency less than 10 out of 3000 sentences) were first removed by the term frequency thresholding method from the keyword list. Based on the above result, chi

square method was more suitable to remove the non-informative keywords in this problem and the details of the experiment were shown below.

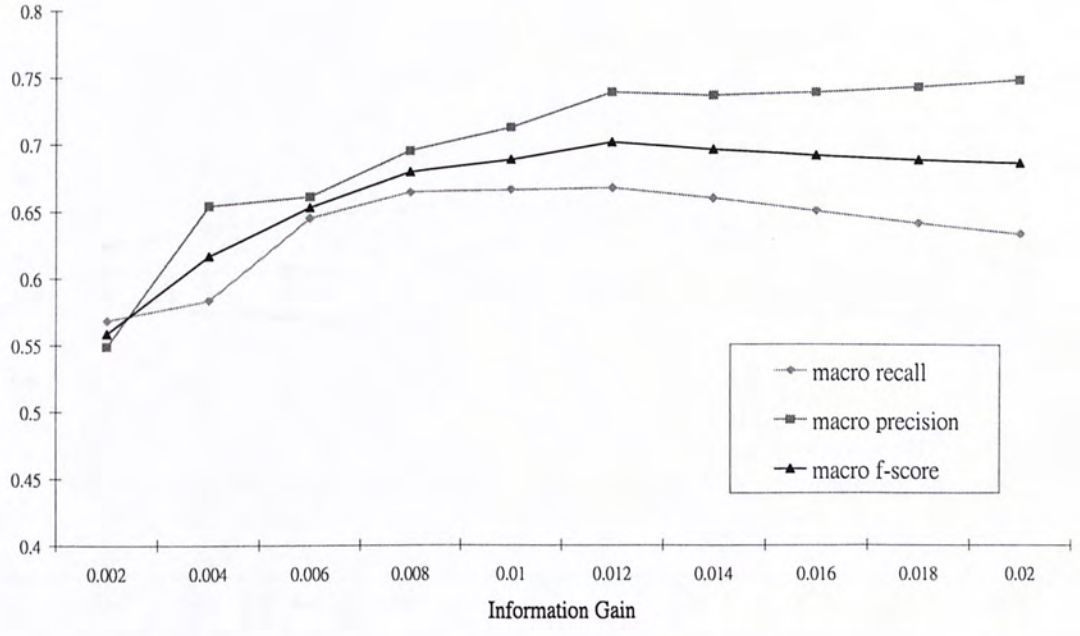

Figure 3.3 Macro F-score value with different information gain value for Bayesian Classifier

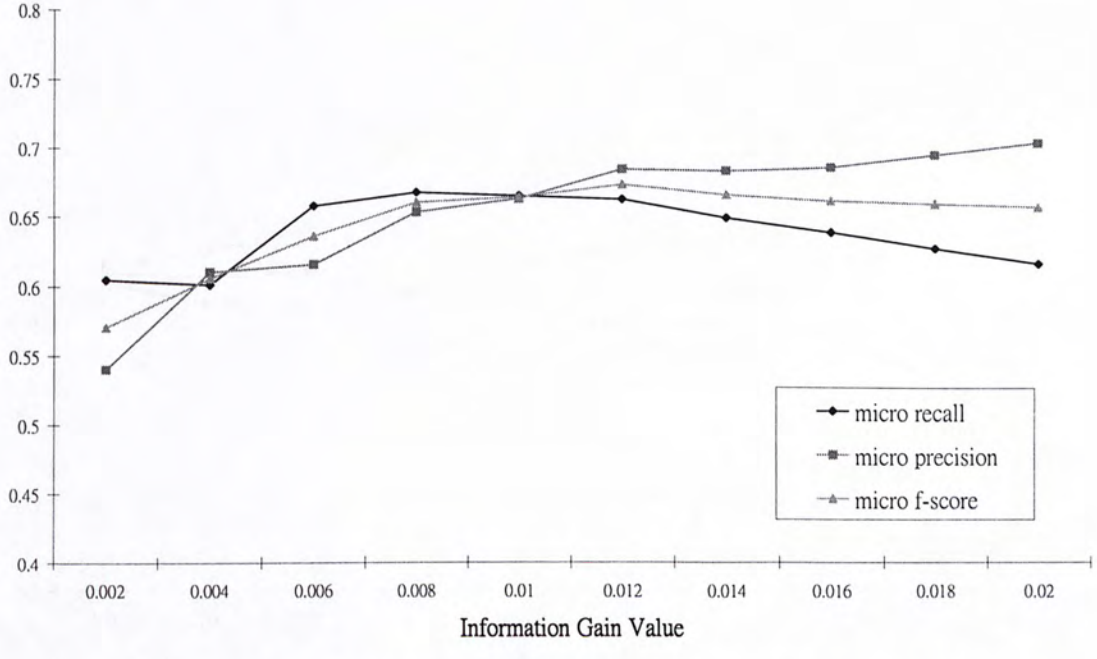

Figure 3.4 Micro F-score value with different information gain value for Bayesian Classifier

Figure 3.3 and 3.4 show the result of Bayesian classier using information gain to remove non-informative keywords. The best macro f-score is 70.19% and 67.19% for the micro f-score with information gain value of 0.012.

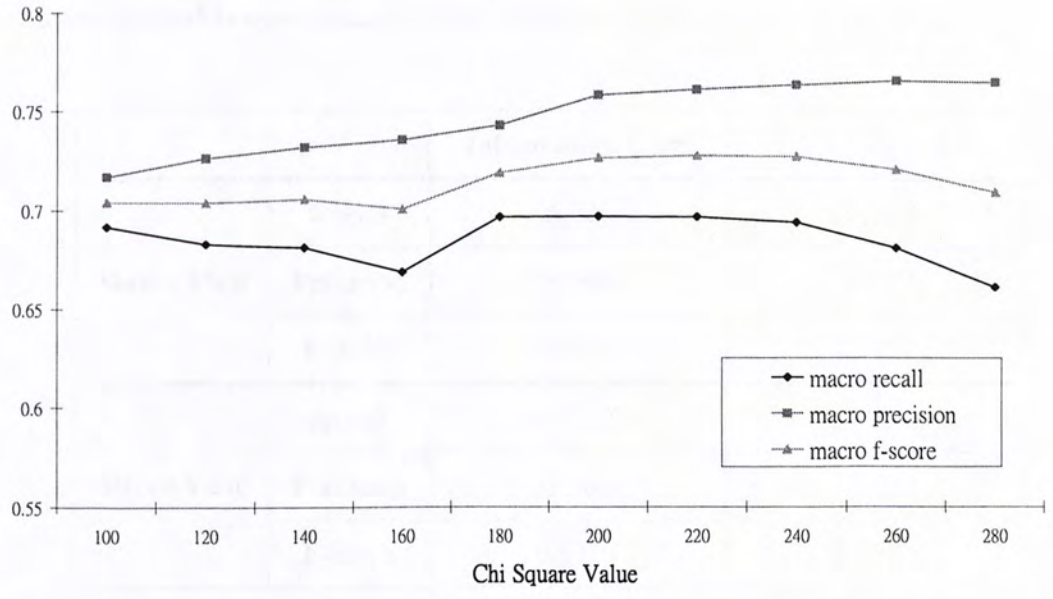

Figure 3.5 Macro F-score value with different chi square value for Bayesian Classifier

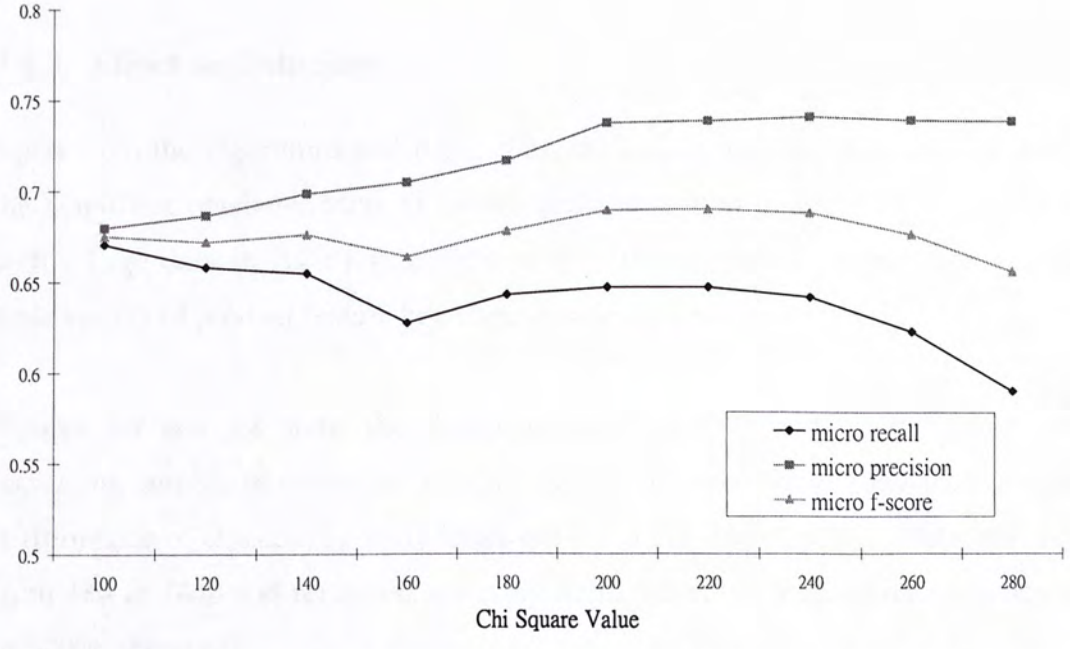

Figure 3.6 Micro F-score value with different chi square value for Bayesian Classifier

Figure 3.5 and 3.6 show the result of Bayesian classier using chi square to remove non-informative keywords. The best macro f-score is 72.77% while the best micro f-score is 68.92% with chi square value 220.

The best result provided by using Information gain and Chi square as the keyword removal method is summarized in the following table.

		Information Gain	Chi Square
Macro View	**Recall**	66.79%	69.70%
	Precision	73.96%	76.12%
	F-score	0.7019	0.7277
Micro View	**Recall**	66.11%	64.70%
	Precision	68.30%	73.80%
	F-score	0.6719	0.6892

Table 3.5 Comparison of the best result provided by the keyword selection method: information gain and chi square

3.4.3 Effect on Data Size

Apart from the algorithms and parameters, the size of training datasets also affect the classifiers obtained. More keywords and association patterns can be extracted with a large dataset. As a result, the quality of the classifiers can be improved if a wide variety of product feature keywords can be captured.

Figures 3.7 and 3.8 show the change of macro f-score and micro f-score with increasing number of sentences in each training set respectively. They show that the performance of classifiers greatly improved when the size of training data increased from 480 to 1440 and remained relatively stable when the training data is made up of >2000 sentences.

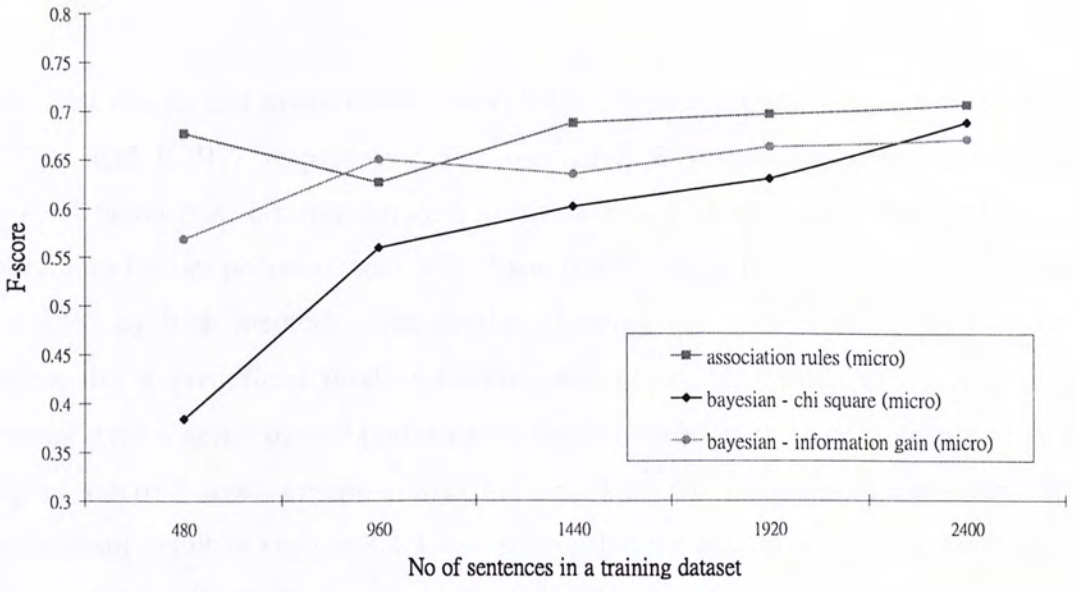

Figure 3.7 Change of macro F-score value with increasing file size for Association Rules Mining

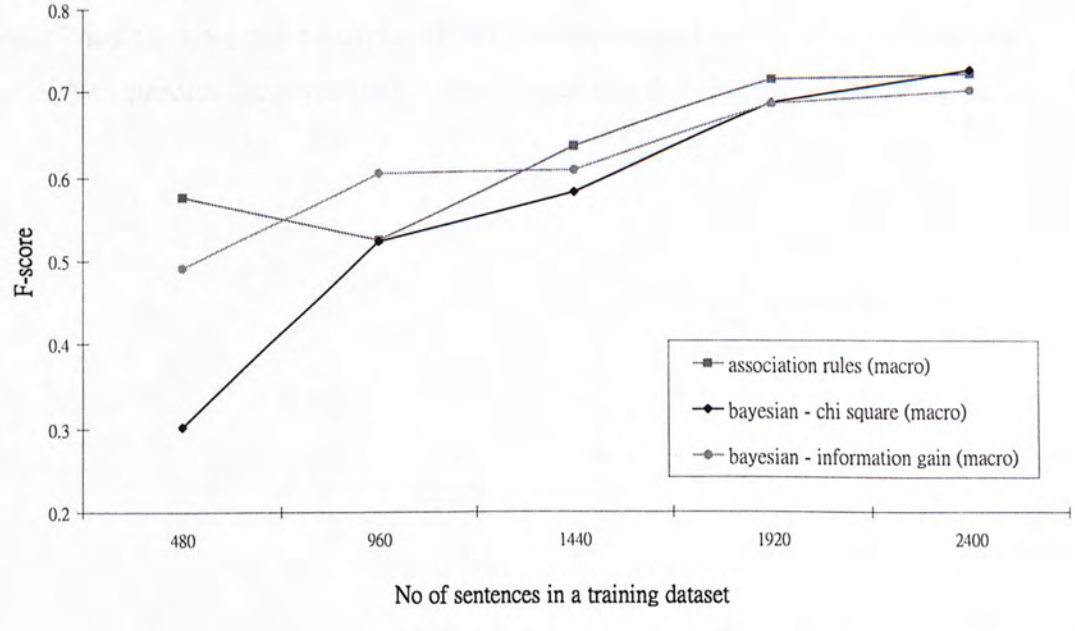

Figure 3.8 Change of micro F-score value with increasing file size for Association Rules Mining

3.5 Discussion

The best macro and micro f-score provided by class association rules mining were 0.7226 and 0.7077 respectively. For the naïve Bayesian classifiers, chi square worked better than information gain in the selection of keywords. The best macro and micro f-score obtained were 0.7277 and 0.6892 respectively. The macro f-scores provided by both methods were similar, showing that their average performance across the 9 predefined product features was closed, but class association rules mining gave a better overall performance for the testing data since it gave a slightly higher micro f-score compare with the result of naïve Bayesian classifier. Our experiment result in section 3.4.3 also shows that the accuracy of class association rules and naïve Bayesian classifier increase as the size of the training sets increase.

Although the results obtained by applying the supervised learning methods were quite promising, there are two major limitations. First, human effort is required. Training sets are needed to build the classifier and predictor. Large amount of human effort is required in order to get a promising result. Second, only the predefined features can be captured. All the supervised learning algorithms can only classify or predict the given data to the classes which appear in the training sets.

4. Extracting Product Feature Sentences Using Unsupervised Learning Algorithms

4.1 Overview

Unsupervised Learning helps to organize data records into similar groups (also known as clusters). The objective of the unsupervised learning algorithms is to minimize the intra-cluster distance and maximize the inter-cluster distance at the same time. Figure 4.1 shows a 2-dimensional data set with 3 natural clusters.

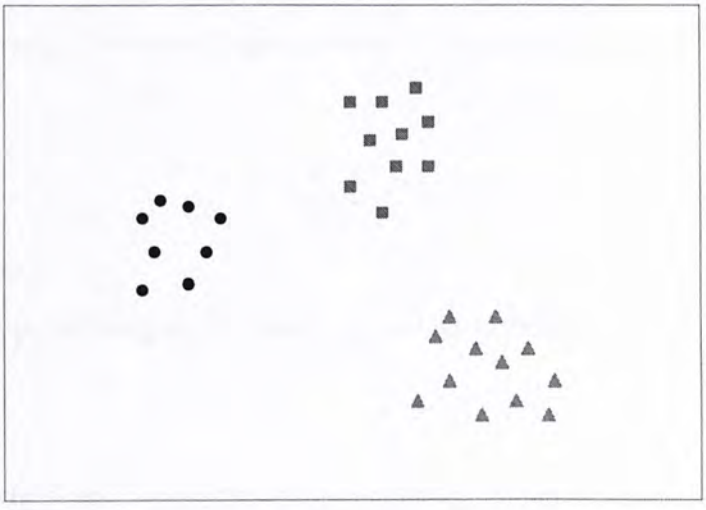

Figure 4.1 Sample Clusters

As shown in the above figure, the clusters formed depend on the distance between the data points only. Unlike supervised learning methods, training datasets with class labels are not needed as an input to produce the classifier. Since the input datasets are not labeled, the clusters found by unsupervised learning algorithms are unlimited to humans' knowledge. Undefined classes can be obtained. These properties can help to solve the problems being faced by supervised learning algorithms stated in section 3.5, i.e. human labeling is needed and only the predefined product features can be captured. In this section, some common unsupervised learning algorithms are being studied.

4.2 Unsupervised Learning Algorithms

4.2.1 K-means Algorithm

K-means algorithm aims to divide a dataset into k clusters. Each cluster is represented by its center which is known as centroid. A centroid is actually the mean of all data in the same cluster.

The algorithm randomly picks k points as the initial centroids. It then computes the distance between the set of centroids and the remaining set of data. Data is assigned to the closest cluster. When the assignment finishes, the centroid of each cluster will be updated. This assignment process iterates until the squared-error criterion function converges. The sum of squared-error, E, for all data in the database is given as follows:

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} dist(p, m_i)^2 \qquad \ldots \qquad (4.1)$$

, where p is the point assigned to cluster C_i, and m_i is the mean of the cluster, i.e. the centroid.

Although this algorithm is simple and fast-running, it can be applied only when the mean can be defined and computed. Since all the data points must be assigned to one of the clusters, it is not suitable to handle noisy dataset. It is also not suitable to deal with non-convex shape clusters as the clusters found by K-mean are always spherical in shape.

4.2.2 Density-Based Scan

Density-Based Scan (DB scan) relies on the notion of density. It believes that the density in the area of noise is lower than that in the area of useful data. Therefore, the algorithm tries to define area with high density as clusters and separate them from low dense noisy area.

The algorithm starts with an arbitrary point p and retrieves all reachable points from p within a predefined radius. If the number of neighbors of the point p exceeds the predefined threshold, *MinPts*, a new cluster is formed and p is regarded as a core point of the cluster while the retrieved points are named as *density-reachable point* from p. The algorithm then examines all the density-reachable points to check if they are also a core point of that cluster. A point q is named as *border point* if it is density-reachable from p but not a core point. Those points which is not within the cluster but reachable from q will be regards as noisy point temporarily. The expansion of that particular cluster will be stopped once all the core points and border points are identified. The assignment process iterates until all the points are either marked with noise or a cluster component.

DB Scan work effectively in discovering cluster with arbitrary shape and filter the noisy data. It performs well in general data, but may not be suitable to cluster data with densities not well-defined.

4.2.3 Hierarchical Clustering

Hierarchical Clustering Algorithm groups data objects into a hierarchy tree structure. It can be subdivided into Agglomerative methods and Divisive methods depending on the strategy used during the construction of tree structure. Agglomerative methods are more commonly used.

Agglomerative hierarchical clustering starts by forming n clusters with each cluster contains one data object. At each stage, two closest clusters are merged until all the data objects are in a single cluster or certain termination conditions are satisfied. Divisive hierarchical clustering does the reverse of agglomerative hierarchical clustering by placing all data objects in a single cluster in the initial stage. The methods then keep splitting the cluster into clusters with smaller size until each object form a single cluster or it meets certain termination conditions. Some possible termination conditions are listed as following:

i) Desired number of clusters is obtained

ii) The distance between two closest clusters exceeds certain predefined threshold for agglomerative methods

iii) The maximum distance between data object of the same cluster is smaller than certain predefined threshold for divisive methods

In hierarchical clustering, the distance between clusters can influence the clusters obtained and their quality. Some commonly used approaches are listed below.

Minimum Distance (also known as Single Linkage Clustering)

The distance between two clusters, C_i and C_j, is represented by the minimum distance between the object pair *(p, q)* where p is in cluster C_i and q is in cluster C_j.

$$dist_{\min}(C_i, C_j) = \min\{dist(p,q) : p \in C_i, q \in C_j\} \quad \ldots \quad (4.2)$$

Maximum Distance (also known as Complete Linkage Clustering)

The distance between two clusters, C_i and C_j, is represented by the maximum distance between the object pair *(p, q)* where p is in cluster C_i and q is in cluster C_j.

$$dist_{\max}(C_i, C_j) = \max\{dist(p,q) : p \in C_i, q \in C_j\} \quad \ldots \quad (4.3)$$

Average Distance (also known as Average Linkage Clustering)

The distance between two clusters, C_i and C_j, is represented by the average distance between all the object pair *(p, q)* where p is in cluster C_i and q is in cluster C_j.

$$dist_{avg}(C_i, C_j) = \frac{1}{|C_i \| C_j|} \sum_{p \in C_i} \sum_{q \in C_j} dist(p,q) \quad \ldots \quad (4.4)$$

4.3 Distance Function

Distance functions play an important role in unsupervised learning. It can influence the clusters obtained. In content clustering, we treat a sentence as a vector and let u be the vector of s_i and v be the vector of s_j in this section.

4.3.1 Euclidean Distance

Euclidean distance is one of the most commonly used distance function. For the two vectors, u and v, with n components, the Euclidean distance is given as follows.

$$Euclidean(s_i, s_j) = \sqrt{\sum_{k=1}^{n} (u_{ik} - v_{jk})^2} \qquad \ldots \qquad (4.5)$$

4.3.2 Jaccard Distance

It measures the dissimilarity between two sentence vectors. It takes the length of the sentences into account and gives a normalized score ranging from 0 to 1. The Jaccard Distance is defined as follows:

$$Jac(s_i, s_j) = 1 - \frac{\sum_{k=1}^{n} u_{ik} v_{jk}}{\sum_{k=1}^{n} (v_{jk})^2 + \sum_{k=1}^{n} (u_{ik})^2 - \sum_{k=1}^{n} u_{ik} v_{jk}} \qquad \ldots \qquad (4.6)$$

4.4 Experiment

In last chapter, we studied the effectiveness of labeling product feature sentences using supervised learning algorithms. Although the classifiers performed satisfactorily in most user-defined product features, some unobvious class liked "red eye reduction", "white balance", etc, remained unlabeled. Therefore, in this section, we used the same set of data described in section 3.4.1 to cluster sentences with similar contents into groups.

4.4.1 Cluster Labeling

The above flowchart shows the cluster labeling process used in this section. Either a cluster with no core meaning or a cluster having >50% of the data not being marked with the major category of that cluster is regarded as "noise".

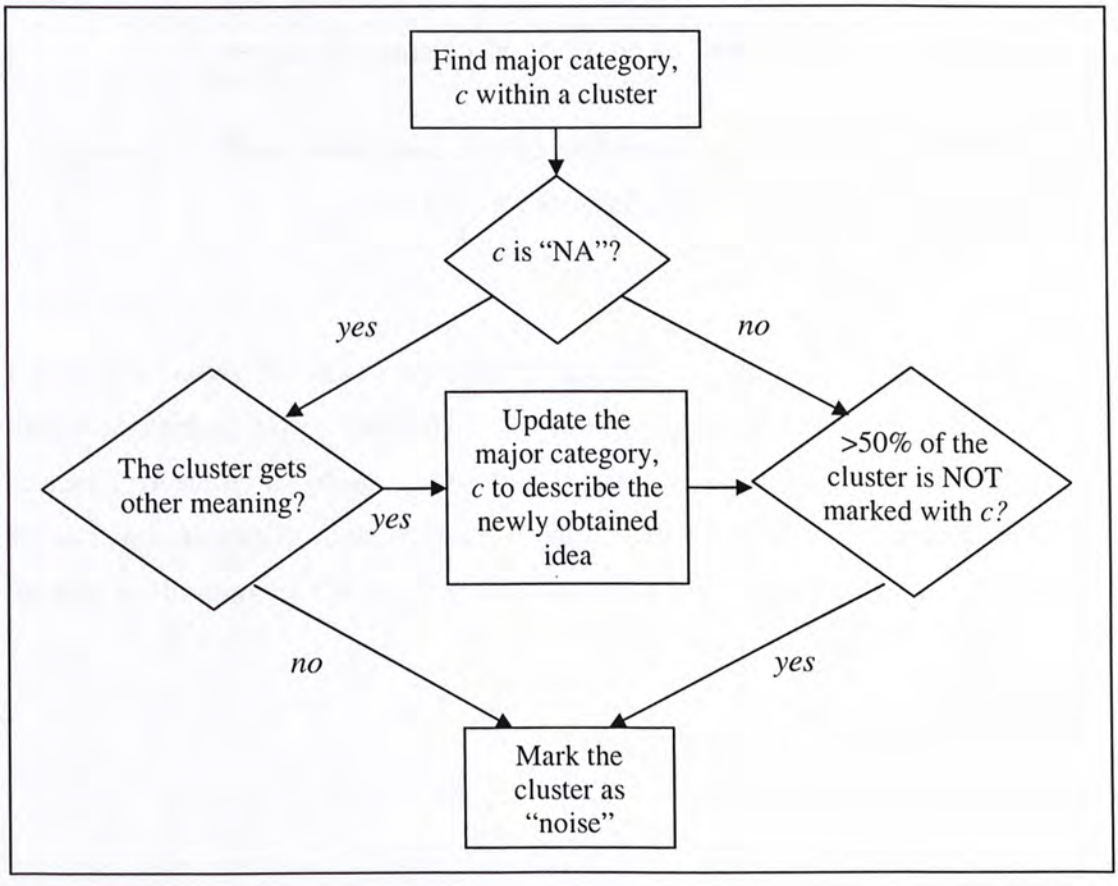

Figure 4.2 Flowchart representing the cluster labeling process

4.4.2 K-means Algorithm

Figure 4.3 compares the total number of clusters and the total number of noisy clusters (with >50% noise) generated by k-means algorithm showing that the algorithm cannot separate the noisy sentences successfully.

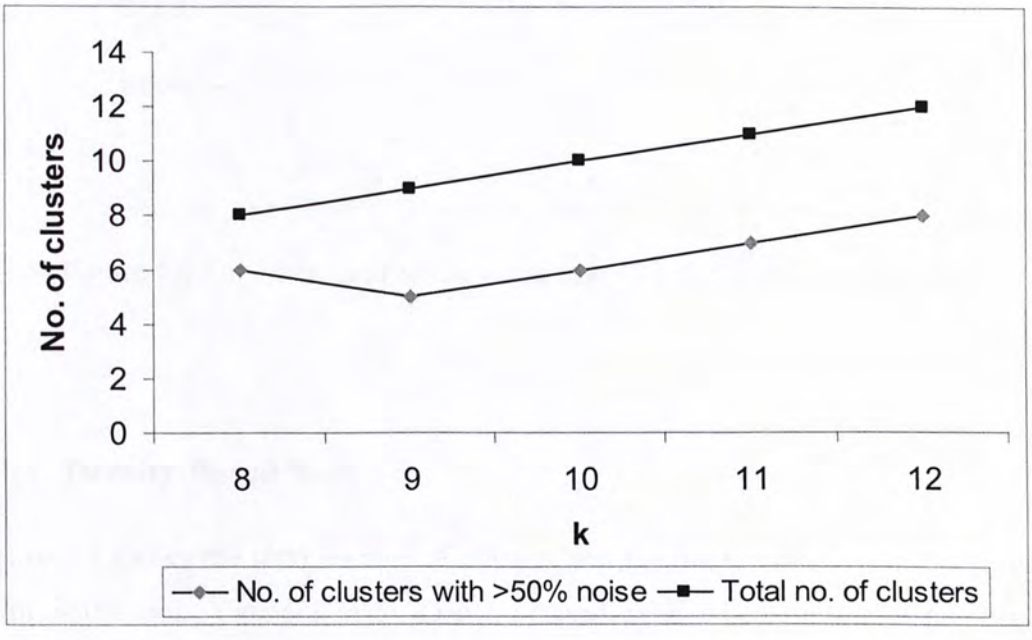

Figure 4.3 Total numbers of clusters and number of clusters with >50% noise VS k for k-means algorithm

Figure 4.4 shows the micro and macro accuracy of the clusters generated by k-means algorithm. Micro accuracy is the total number of sentences that have been correctly clustered dividing by the total number of sentences have been clustered while macro accuracy is the average of the number of correctly clustered sentences in cluster *i* dividing by the number of sentence in cluster *i* for all cluster.

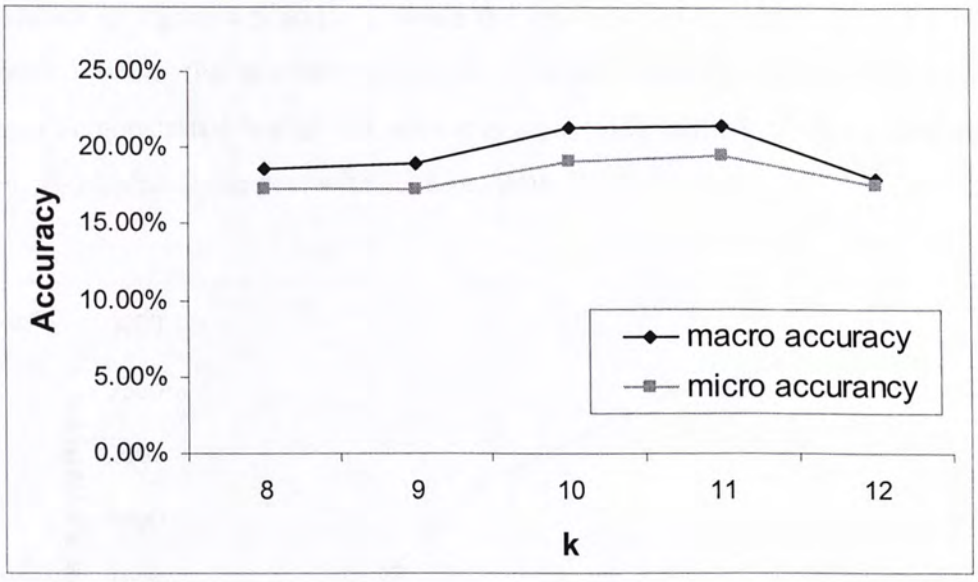

Figure 4.4 Micro and macro accuracy VS k for k-means algorithm

4.4.3 Density-Based Scan

Figure 4.5 shows the total number of clusters and the total number of noisy clusters (with >50% noise) generated by Density-Based Scan when minimum number of points = 3. All the clusters contained over 50% noisy sentences.

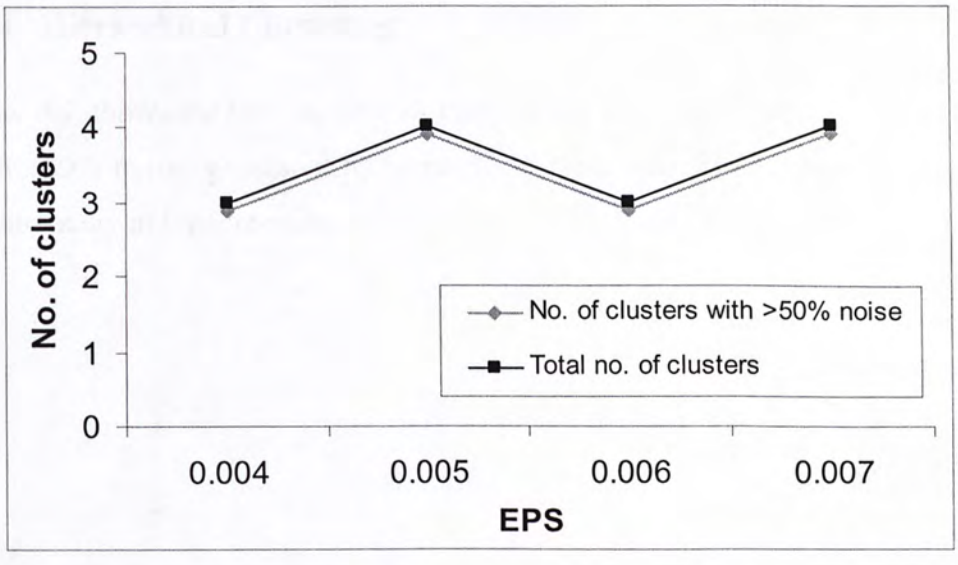

Figure 4.5 Total number of clusters and number of clusters with >50% noise VS eps for DB Scan algorithm when minpts = 3

As shown in figure 4.5 and 4.6, when the *eps* was loosen, the size of the clusters increased while the number of clusters did not have significant changes. Such pattern demonstrated that all the sentences were lying in the high dense area and DB Scan algorithm cannot deal with such problem.

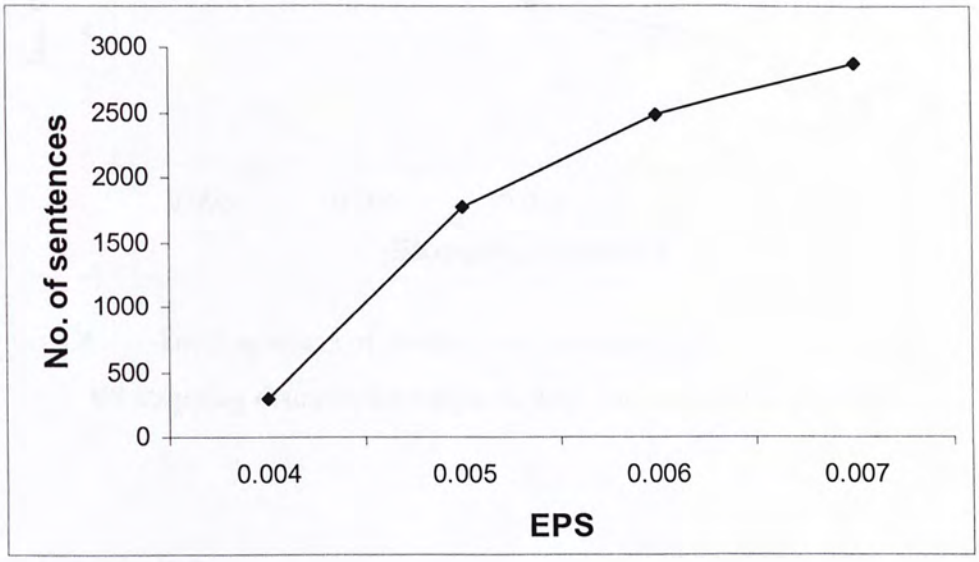

Figure 4.6 Number of sentences have been clustered for DB Scan algorithm when minpts = 3

4.4.4 Hierarchical Clustering

Figure 4.7 shows the total number of clusters and the total number of noisy clusters (with >50% noise) generated by hierarchical clustering. The clusters obtained also contain many noisy sentences.

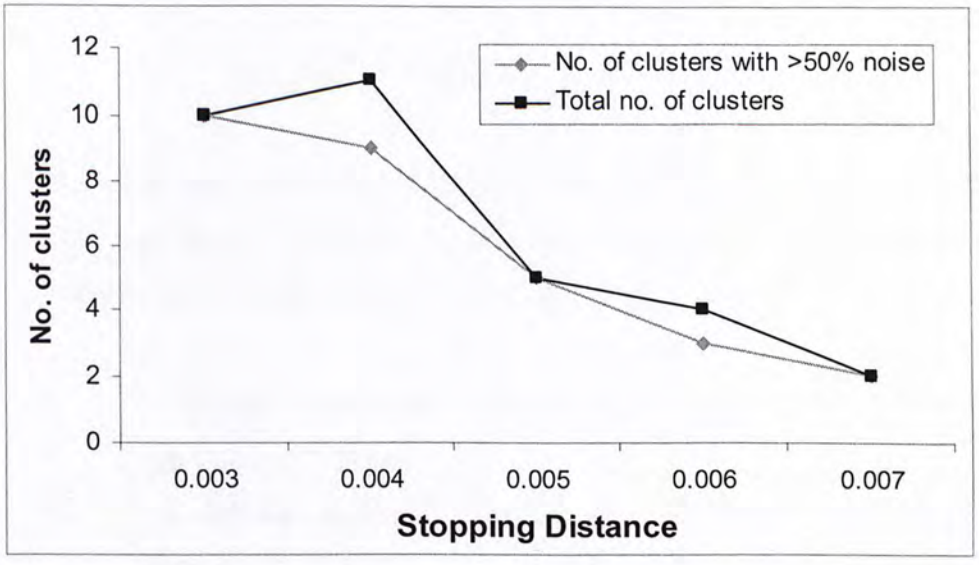

Figure 4.7 Total numbers of clusters and number of clusters with >50% noise VS stopping distance for single linkage hierarchical algorithm

4.5 Discussion

In this work, sentences were used as the basic unit. Since sentences are made up of a few words only, it provides less information compare with documents which contain a wide variety of words and combinations. Also, several synonyms can be used to describe the same feature, for example, "video" is equivalent to "movie" but they seldom appear in the same sentence. Therefore, it is difficult to obtain an accurate measure for the distance between the sentences. Consider the following three sentences:

S_1: *"The LCD is bright and glossy."*

S_2: *"This camera has a high quality screen."*

S_3: *"This camera can capture high quality picture."*

It is obvious that both S_1 and S_2 are describing the same product feature '*LCD screen*', but the distance between them are zero no matter what distance function is used. Although S_3 comments on the product feature '*picture quality*', it shares some common words with S_2. The comparison between their distances is shown as below.

$$dist(S_2, S_3) > dist(S_1, S_2) = dist(S_1, S_3)$$

In addition, most web users would like to tell their story instead of comment on the product features directly. When we decompose a review into a stream of sentences, many of them may not carry any product feature. S_4 and S_5 are two typical examples.

S_4: "I bought this camera to replace an Olympus c 2000 2.1 mp which I gave to my parents."

S_5: "I only had a couple of days to get used to the controls before we departed."

The removal of the noisy sentences is another important task in the extraction of product feature from online reviews. Hence, sentence clustering is not suitable in tackling the problem in this work.

5. Extracting Product Feature Sentences Using Concept Clustering

5.1 Overview

According to chapters 3 and 4, the performance of supervised learning algorithms was much better than that of unsupervised learning algorithms in the extraction of product feature sentences. The drawback is that human effort is needed and only the predefined product feature sentences can be identified. For unsupervised learning algorithms, the major problem is that it is strongly relied on the similarity between the sentences rather than the occurrence of specific keywords in the sentences. In this section, we will introduce the idea of "Concept Clustering", which is able to overcome the weaknesses of both supervised and unsupervised learning algorithms.

Concepts Clustering refers to organizing terms that describe the same idea (also known as concept) into group. A set of sentences is acted as an input and the distinct terms within the set of sentences are used as the basic unit in forming the concept clusters. The similarities between the terms are collected based on their co-occurrence frequency in the input set of sentences. Different unsupervised learning techniques can be used to cluster the terms according to the similarities obtained statistically. Since the terms in the same cluster can describe the same concept, a classifier can then be built to extract sentences which depict that particular concept.

In this chapter, we employ the existing Density-based Scan algorithm and Hierarchical clustering algorithm to study the effectiveness of concept clustering in the extraction of product features from online consumer reviews.

5.2 Distance Function

In concept clustering, we need to compute the distance between two terms, let t_j and t_k be any two terms in the set of words, T.

5.2.1 Association Weight

The associated term weight (Li & Yang, 2005) measures the relevance weights from one term t_j to the other term t_k. It divides the co-occurrence weight between t_j and t_k by the occurrence frequency of t_j. The weighting factor is used to penalize some general terms. The formulation is given as follows.

$$Weight(t_j, t_k) = \frac{\sum_{i=1}^{n} d_{ijk}}{\sum_{i=1}^{n} d_{ij}} \times WeightingFactor(t_k) \quad \ldots \quad (5.1)$$

, where

$$d_{ijk} = tf_{ijk} \times \ln(\frac{N}{df_{jk}}) \quad \ldots \quad (5.2)$$

$$d_{ij} = tf_{ij} \times \ln(\frac{N}{df_j}) \quad \ldots \quad (5.3)$$

$$WeightingFactor(t_k) = \frac{\ln \frac{N}{df_k}}{\ln N} \quad \ldots \quad (5.4)$$

, and

tf_{ij} is the occurrence frequency of t_j in document i

df_j is the number of sentences containing t_j

tf_{ijk} is the minimum between occurrence frequency of t_j and t_k in sentences i

df_{jk} is the number of sentences containing t_j and t_k

N is the total number of sentences

5.2.2 Chi Square

It evaluates statistically significant differences between term t_j and term t_k (Yang & Pedersen, 1997). It has a natural zero if the terms are independent. The Chi-Square is defined as follows:

$$\chi^2(t_j, t_k) = \frac{N\left[P(t_j \wedge t_k)P(\bar{t}_j \wedge \bar{t}_k) - P(t_j \wedge \bar{t}_k)P(\bar{t}_j \wedge t_k)\right]^2}{P(t_k)P(\bar{t}_k)P(t_j)P(\bar{t}_j)} \quad \ldots \quad (5.5)$$

5.2.3 Mutual Information

Mutual Information measures the mutual dependence of a term and a class (Yang and Pedersen, 1997). The formula of the mutual information between a term t and a class c is given as follows:

$$I(t,c) = \log \frac{P(t \wedge c)}{P(t)P(c)} \quad \ldots \quad (5.6)$$

5.3 Experiment

In this section, we employed Density-based Scan algorithm and Single-linkage Hierarchical clustering algorithm studied in last chapter to do the concept clustering. The set of digital camera sentences described in section 3.4.1 was used to extract the sets of product feature related terms. The terms were then used to extract the product feature sentences.

5.3.1 Effect on Distance Functions

The distance between two terms was normalized to a value between 0 and 1. Min-max normalization is used and the formula is given as below.

$$v' = \frac{v - \min_s}{\max_s - \min_s} \qquad \ldots \qquad (5.7)$$

, where v' is the normalized value, v is the original value, max_d and min_d are the maximum and minimum value among the set of distance respectively.

We evaluated the effectiveness of the three distance functions discussed in section 5.2 by measuring the percentage of valid clusters with respect to various total number of clusters extracted with DB Scan and Hierarchical clustering algorithm. The results are shown in figure 5.1 and 5.2 respectively. A cluster was regarded as valid if it related to a concept about a product feature. Both figures show that Chi Square was better than the Mutual Information as well as the Association Weight in calculating the distance between the terms. Hence, only Chi Square was used as the distance function in the remaining sections.

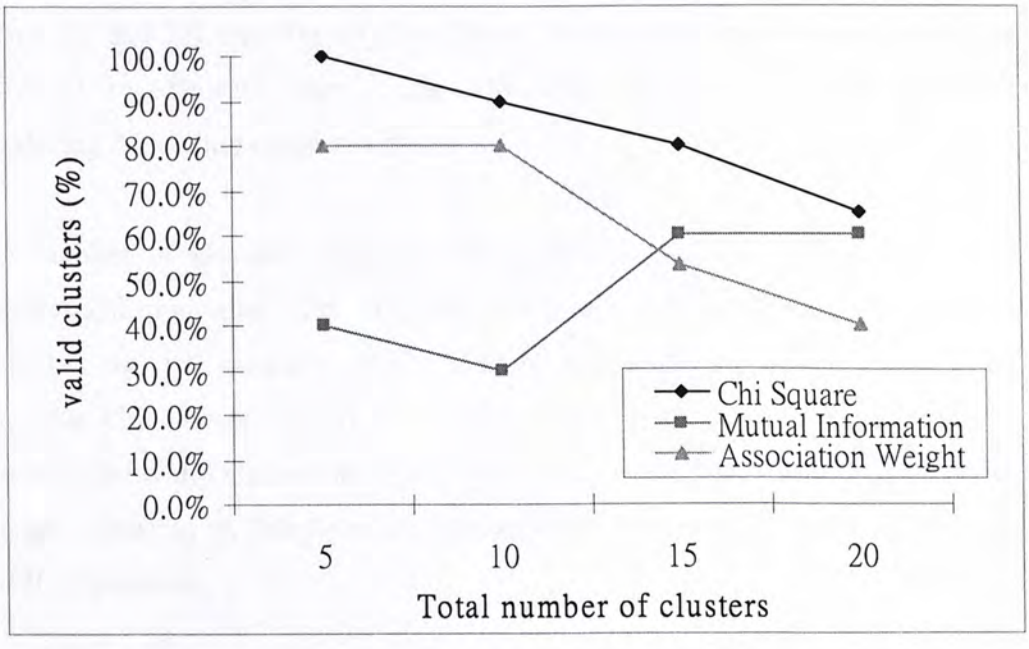

Figure 5.1 Percentage of valid clusters extracted by DB Scan Algorithm with different distance function

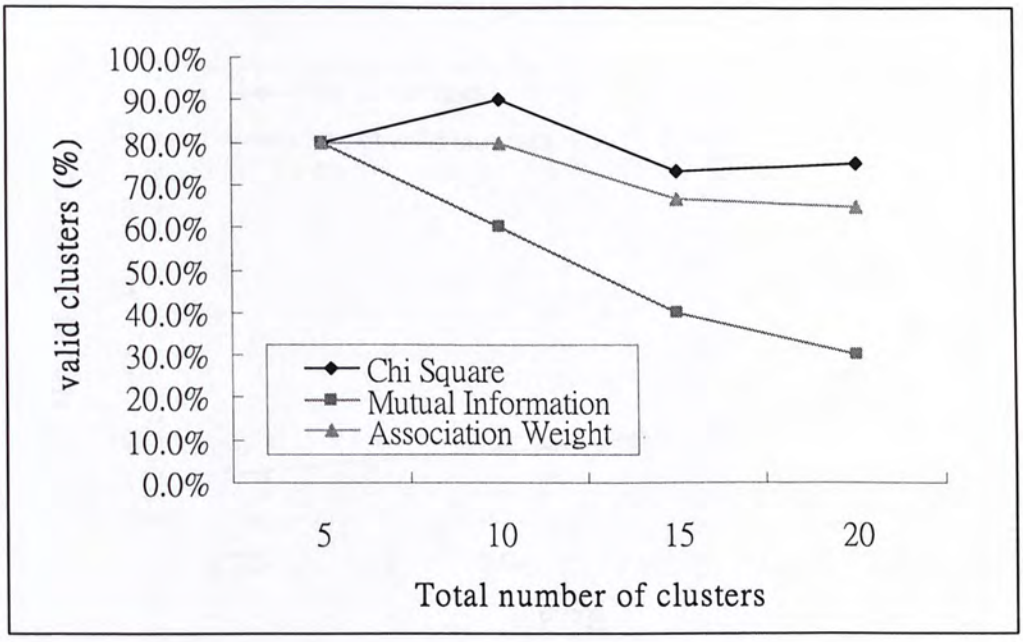

Figure 5.2 Percentage of valid clusters extracted by Single Linkage Hierarchical Clustering Algorithm with different distance function

5.3.2 Extraction of Product Features Clusters

Figure 5.3 and 5.4 show the total number of clusters and the number of valid clusters extracted by Density-based Scan Algorithm and Single-linkage Hierarchical Clustering Algorithm respectively.

The number of clusters found by DB Scan was much less than that found by Hierarchical clustering since DB Scan could only capture concepts with at least 3 related terms, i.e. minPts = 3. For Hierarchical clustering, it was able to capture concepts which were related to 2 terms only, but the disadvantage was that the numbers of invalid clusters increase significantly at the same time. The details of the clusters extracted by DB Scan and Hierarchical clustering are listed in Appendix A and II respectively.

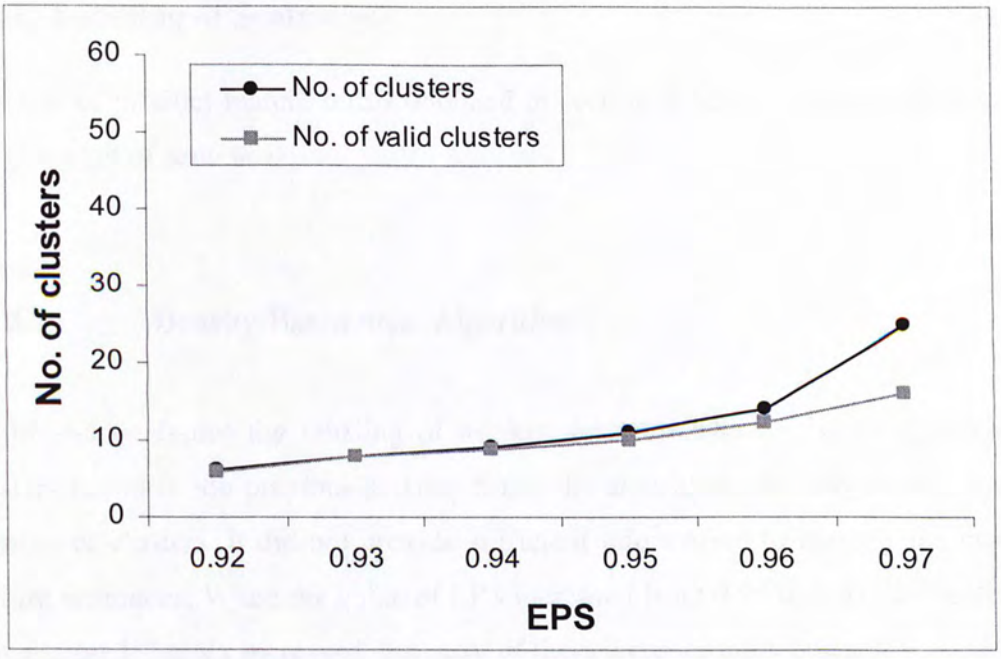

Figure 5.3 No of valid clusters and the total number of clusters extracted by DB Scan Algorithm when minpts = 3 and EPS ranged from 0.92 to 0.97

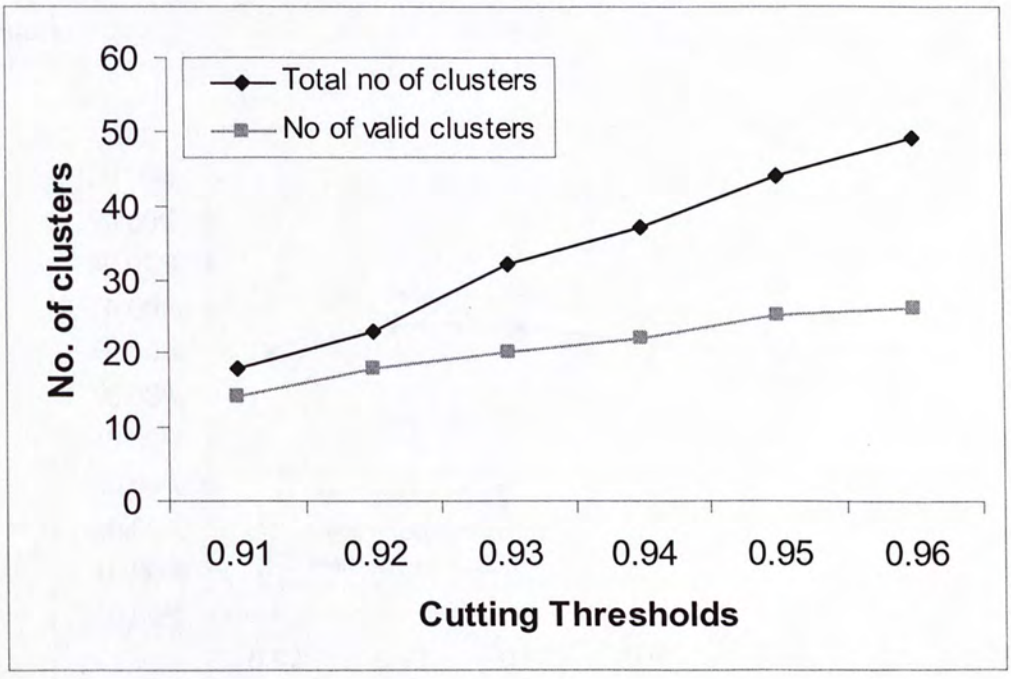

Figure 5.4 No of valid clusters and the total number of clusters extracted by Hierarchical Clustering Algorithm with different cutting thresholds

5.3.3 Labeling of Sentences

The sets of product feature terms obtained in section 5.3.2 were further used to tag the given set of sentences.

5.3.3.1 Density-Based Scan Algorithm

Figure 5.5 evaluates the labeling of product feature sentences by using the valid clusters found in the previous section. Since the algorithm can only extract a small number of clusters, it did not provide sufficient information to identify the product feature sentences. When the value of EPS increased from 0.96 to 0.97, the sentences being tagged sharply increased, but most of them were wrongly labeled.

Figure 5.6 further evaluates the performance of the valid clusters with macro point-of-view. The overall performance of the clusters remained steadily with macro F-score ranging from 70.8% to 68.5% for EPS = 0.92 to 0.96. When EPS = 0.97, the precision dropped suddenly showing that so noise may be included in the valid clusters.

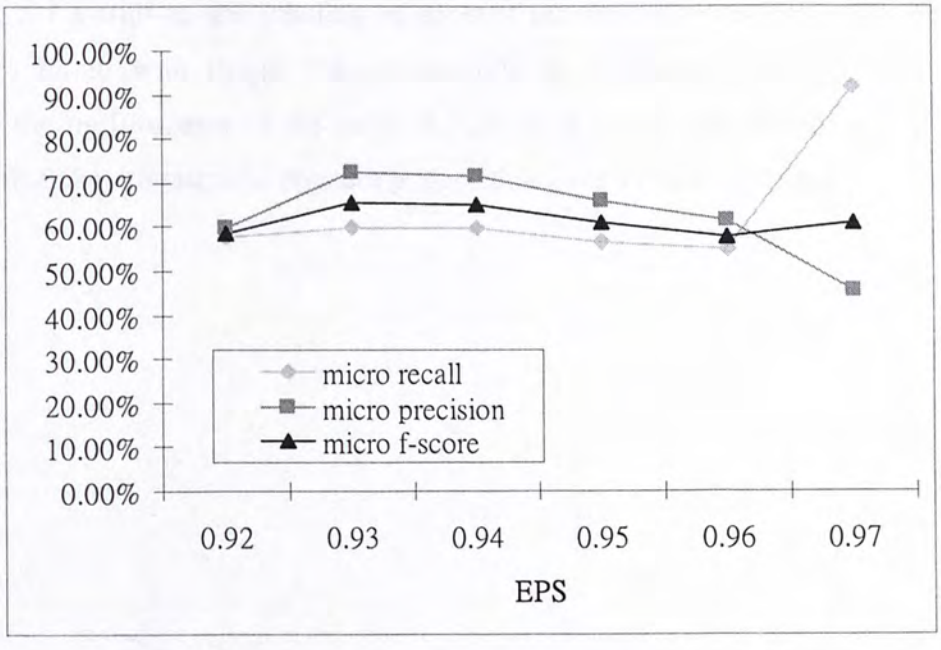

Figure 5.5 Micro views of the labeling of product feature sentences by using the valid clusters found by DB Scan Algorithm

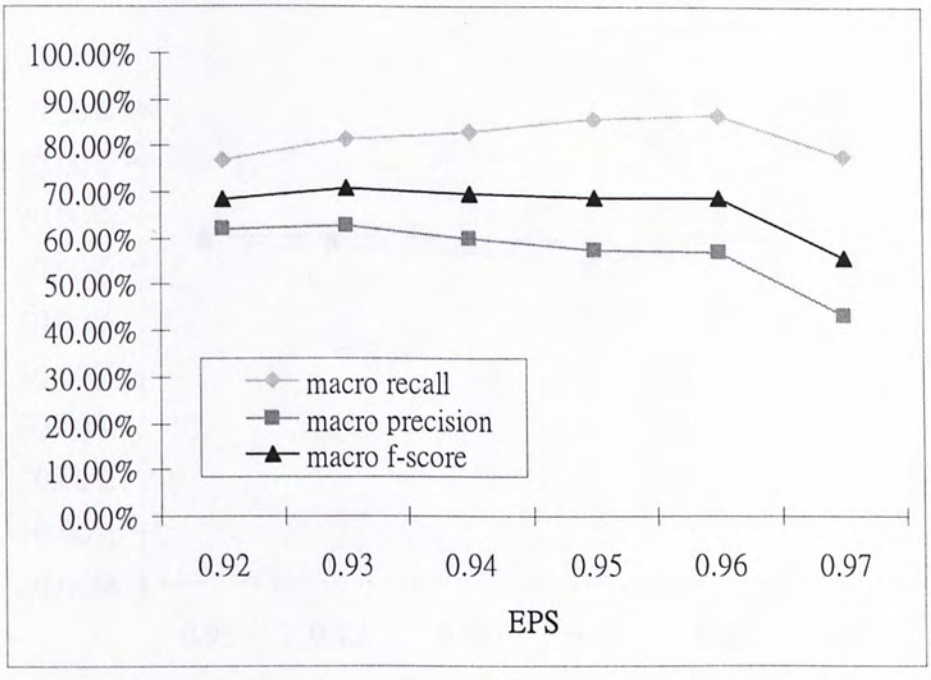

Figure 5.6 Macro views of the product feature clusters found by DB Scan Algorithm with minPts = 3 and different EPS

5.3.3.2 Single-linkage Hierarchical Clustering Algorithm

Figure 5.7 evaluates the labeling of product feature sentences by using the valid clusters found with single linkage hierarchical clustering algorithm. Figure 5.8 shows the performance of the valid clusters with macro point-of-view. Compared with DB Scan, hierarchical clustering algorithm gave a more stable result.

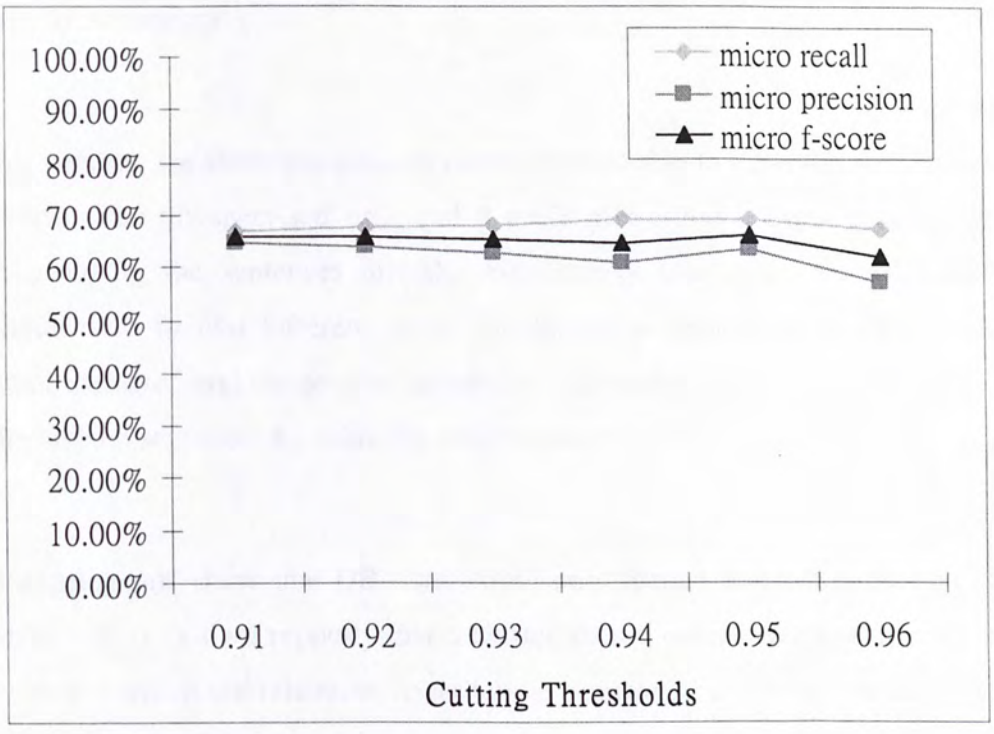

Figure 5.7 Micro views of the labeling of product feature sentences by using the valid clusters found by Hierarchical Clustering Algorithm

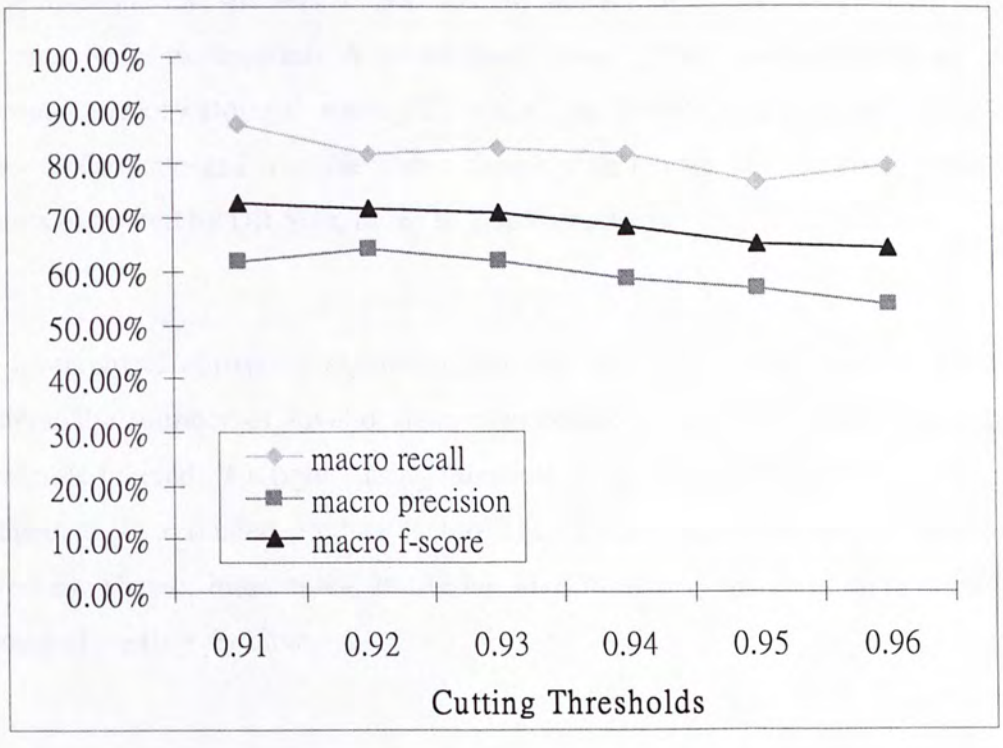

Figure 5.8 Macro views of the product feature clusters found by Hierarchical clustering Algorithm with different cutting threshold

5.4 Discussion

In this chapter, we show that concept clustering was able to label the product feature sentence in an unsupervised style and it could give a much better result compare with grouping the sentences directly. For concept clustering, we evaluated the methods used in two different ways. Firstly, we measured the number of valid clusters extracted and the relative percentage. Secondly, we evaluated the accuracy of the labeled sentences by using the valid clusters.

Our experiments show that DB Scan could only extract a small number of valid clusters due to its own property that a cluster should contain no less than 3 points. For clusters which are related to fewer terms, it may no be able to extract. Take the product feature "LCD screen" as an example, it is related to "LCD" and "screen" in common situation. Even DB Scan able to group the terms "LCD", "screen" and "view" in a single cluster (refer to Appendix A), the term "view" is relatively common and hence may affect the labeling of sentences. In addition, when the EPS was loosened, the clusters might contain much noise and merge two distinct concepts. Refer to Appendix A, terms liked, "door", "free", etc were grouped to the concept "memory storage" when EPS = 0.97 and two other concepts, "video" and "auto mode", merged into the same cluster with cluster ID 13. As a result, the clusters extracted by DB Scan failed to give correct label to the sentences.

For hierarchical clustering algorithm, although it could extract most of the valid clusters, the number of invalid clusters increased more rapidly when the cutting thresholds relaxed. If a tight cutting threshold is set, some useful terms may not be grouped to their related product feature clusters and finally affect the labeling of sentences. Hence, hierarchical clustering algorithm was not best fit to tackle the concept clustering problem.

6. Extracting Product Feature Sentences Using Concept Clustering and Proposed Unsupervised Learning Algorithm

6.1 Overview

In previous chapter, we have studied the extraction of product feature sentences using concept clustering with existing unsupervised learning algorithms. It shows that both density-based and hierarchical clustering methods are not best-fit to the problem addressed in this work as density-based algorithm cannot extract product feature clusters which are related to less than three keywords and hierarchical algorithm extracts a lot of noisy clusters.

Since a product feature may associate with various keywords and different keywords can have different association strengths to the product feature. In this chapter, we present our proposed Scalable Distance Clustering Algorithm which is able to extract concept clusters with the indication of their members' strength to the concepts. A threshold function is acted as the input to the algorithm and the formation and expansion of a concept clustering is restricted by the corresponding thresholds value with respect to the cluster size.

In section 6.2, we formally define the problem. Section 6.3 introduces the details of the Scalable Distance Clustering algorithm. Section 6.4 discusses the properties of the algorithm. In section 6.5, various experiments are conducted. Lastly, we discuss the overall performance and effectiveness of the Scalable Distance Clustering Algorithm.

6.2 Problem Statement

Let $R_p = \{r_{p\_1}, r_{p\_2}, \ldots, r_{p\_n}\}$ be a set of opinions (also known as reviews) of a particular type of product p (which may be from different brands and different models). Let $T_p = \{ t_1, t_2, \ldots, t_n \}$ be a set of terms, such that $r_i \subseteq T$. Let $C_p = \{c_{p\_1}, c_{p\_2}, \ldots, c_{p\_m}\}$ be a set of concept clusters of a particular type of product p found by concept clustering. Concept clustering refers to the extraction of terms which are related to a concept cluster, such that the terms can be used to identify the related concept, i.e. $\{ t_{p\_1\_1}, t_{p\_1\_2}, \ldots, t_{p\_1\_k} \} \rightarrow c_{p\_1}$.

Definition 1 (Threshold Function)

A threshold function, *F_thres*, is a user-defined function which is used to determine the threshold value with respect to the size of the clusters. A point q can be marked as a member of cluster C if and only if the average distance between q and all the members of C is lower than the corresponding threshold function value, $F\_thres(|C|)$, i.e. $\frac{1}{|C|}\sum_{p \in C} dist(p,q) \leq F\_thres(|C|)$

Definition 2 (Layer)

A layer is a subset of cluster, i.e. $l_i \subseteq C$ for all i. All points from the same layer are obtained from the same iteration with the same threshold function value. Terms in the same layer should have the same strength in the identification of the cluster.

6.3 Proposed Algorithm – Scalable Thresholds Clustering

The algorithm takes the input threshold function and works with two major steps. It identifies the possible set of initial seeds for the clusters and sorts them according to their distances in ascending order. The algorithm then expands the unclassified

initial seeds with respect to their own size and the predefined threshold function. The algorithm is summarized in this section.

```
SCALABLE (Points, F_thres)

// Points is the set of data points
// F_thres is an object to obtain the initial threshold
   // and the expanding thresholds


// Find the possible set of clusters
  i_thres := F_thres.get_threshold(1);  // Get the initial threshold
  FOR i FROM 1 to Points.size-1 DO
    pi := Points.get(i);
    FOR j FROM i+1 to Points.size DO
      pj := Points.get(j);            // Compute the distance of each
      dist := distance(pi, pj);                    // pair of points
      IF (dist < i_thres)       // Save those pairs with distance less
        PairList.insert(dist, pi, pj);   // than i_thres to PairList
      END IF
    END FOR
  END FOR

// Retrieve all clusters
  cid := 1;
  WHILE PairList <> Empty DO
    pair := PairList.pop();
    p1 := pair.getPoint(1);
    p2 := pair.getPoint(2);

  // Check if the pairs of points can form a new cluster
    IF (p1.CID = UNCLASSIFIED) AND (p2.CID = UNCLASSIFIED) THEN
      clust := Clusters.newCluster(cid);     // Create a new cluster
                                                  // with a new cid
      layer := clust.newLayer(); // Create a new layer of the cluster
      clust.addPoint(layer, p1);       // Add p1 and p2 to the cluster
      clust.addPoint(layer, p2);
      p1.CID = cid;             // Assign the current cid to p1 and p2
      p2.CID = cid;

    // Further expand the cluster
      REPEAT
        k := min(clust.getClusterSize(), F_thres.size);
        e_thres := F_thres.get_threshold(k);
      UNTIL ExpandCluster(Points, clust, k, e_thres, cid) = False;
      cid := cid + 1;
    END IF
  END WHILE

END;
```

F_thres is the input threshold function with respect to the size of cluster and the method *F_thres.get_threshold(i)* returns the corresponding threshold value with cluster size *i*. To find the possible set of initial seeds, the algorithm

extracts the first value of the threshold function, *i_thres*. If a pair of points whose distance, *dist*, is shorter than *i_thres*, the points will become one of the possible initial seeds and being inserted by the function *PairList.insert(dist, i, j)*. The pair list is inserted according to the pair-wise distance in ascending order.

Whenever a pair of points is not yet classified, a new cluster is formed. *Clusters* is a collection of cluster objects and the function *Clusters.newCluster(cid)* returns a new object, *clust*, with *cid* as the cluster ID. The method *clust.newLayer()* is then called to create a new layer for *clust* and the corresponding layer ID, *layer*, is returned. The pair of points represents the most inner layer of the cluster, i.e. layer 1. The method *clust.addPoint(layer, p)* helps to locate the point *p* to the cluster with the input layer ID.

The expansion process repeats until no layer is further created. In each iteration, the threshold value is updated according to the size of cluster, which is found by the method *clust.getClusterSize()*. Since the cluster may grow exponentially and the number of threshold value, i.e. *F_thres.size*, may be limited, it is necessary to take the minimum between the cluster size and number of threshold value for choosing a suitable threshold value. The most important method for the expansion of clusters is shown below.

```
ExpandCluster(Points, clust, k, e_thres, cid):Boolean;

// Search for the Points that is reachable from cluster c

  c_Points := clust.getAllPoints();
  shortest_k := Arrays.newArray(k);  // Declare a new array of size k
  FOR i FROM 1 to Points.size DO
    pi := Points.get(i);
    IF (pi.CID = UNCLASSIFIED)
      FOR j FROM 1 to c_Points.size DO
        dist := distance(pi, c_Points[j]);
        Arrays.SortInsertion(shortest_k, dist);
      END FOR
      IF (Arrays.average(shortest_k) < e_thres)
        expand.add(i);                    // Add pi to the expand list
      END IF
    END IF
  END FOR
```

```
// Edit the Clusters and Points
  IF expand = Empty THEN
    RETURN False;
  ELSE
    layer := clust.newLayer();
    WHILE expand <> Empty DO
      p_next := expand.pop();  // Extract points from the expand list
      p_next.CID = cid;          // Assign cid to the extracted pints
      clust.addPoint(layer, p_next); // Add extracted points to layer
    END WHILE
  END IF
  RETURN True;

END;
```

The expansion process starts with measuring the average distance between an unclassified point p and its k nearest neighbors from `clust` and it can be formulated as follows:

$$k\text{-NN dist}(p, \texttt{clust}) = \frac{\sum_{i=1}^{k} \textit{the i-th shortest distance between p and points in } \texttt{clust}}{|k|}$$

The algorithm first extracts the set of points in the cluster by using the function `clust.getAllPoints()` and creates an array with size of k to store the k-shortest distance between a point p and the points in `clust` by using `Arrays.newArray(k)`. `Arrays.SortInsertion(shortest_k, dist)` helps to insert `dist` to the array `shortest_k` and sort the array in ascending order at the same time. The method `Arrays.average(shortest_k)` calculates the average value of all the elements of an array, i.e. k-NN dist (p, `clust`). If the k-NN distance between is shorter than the input parameter, `e_thres`, the point p will be put into the set *expand*. If the set is not empty, a new layer is created and all the points in *expand* become the member of the corresponding layer of the cluster `clust`.

6.4 Properties of the Proposed Unsupervised Learning Algorithm

In this section, we discuss various properties of the Scalable Distance Clustering Algorithm.

6.4.1 Relationship between threshold functions & shape of clusters

When a cluster expands, the input parameter for the threshold function is either bounded by the size of cluster, `clust.getClusterSize()`, or the size of threshold function, `F_thres.size`. If the size of threshold function is large enough as shown in figure 5.1, the input parameter of threshold value is controlled by the size of cluster only. Hence, the distance between a new coming point, p, and a cluster is determined by the average distance of between p and all the points in a cluster. The resultant cluster formed should be with convex shape which is similar to the one shown in figure 5.2.

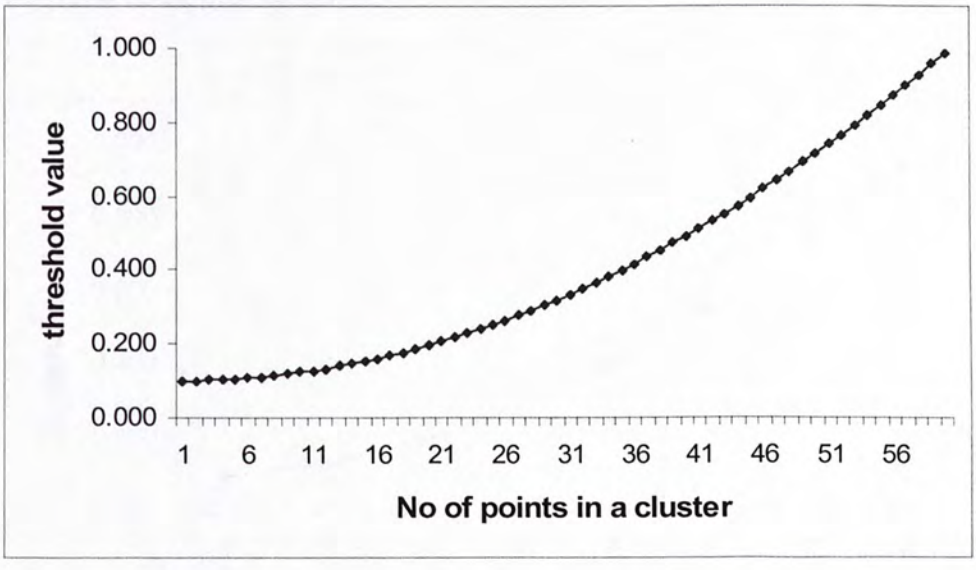

Figure 6.1 Threshold Function with large size

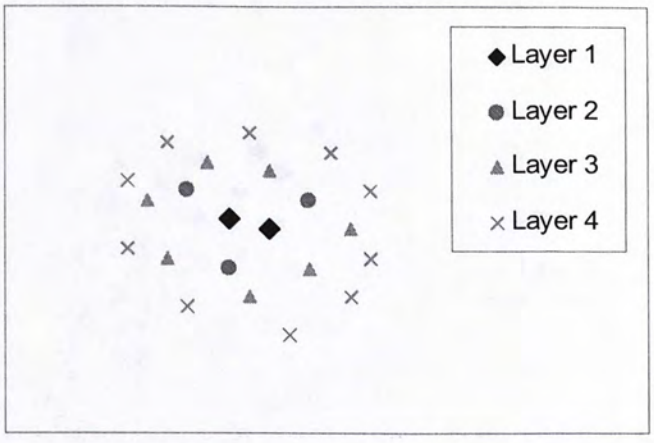

Figure 6.2 Convex-shape Cluster built with large threshold size

On the contrary, if the size of threshold function is too small as shown in figure 5.3, the input parameter of the threshold value is always controlled by the size of the threshold function when the cluster size exceed the function size. For this case, the expansion of a cluster is determined by part of the existing cluster only which is similar to the expansion process of Density-Based Algorithm. The result cluster formed may be with elongated shape and the two ends of a cluster may be loosely held by some weak linkage points.

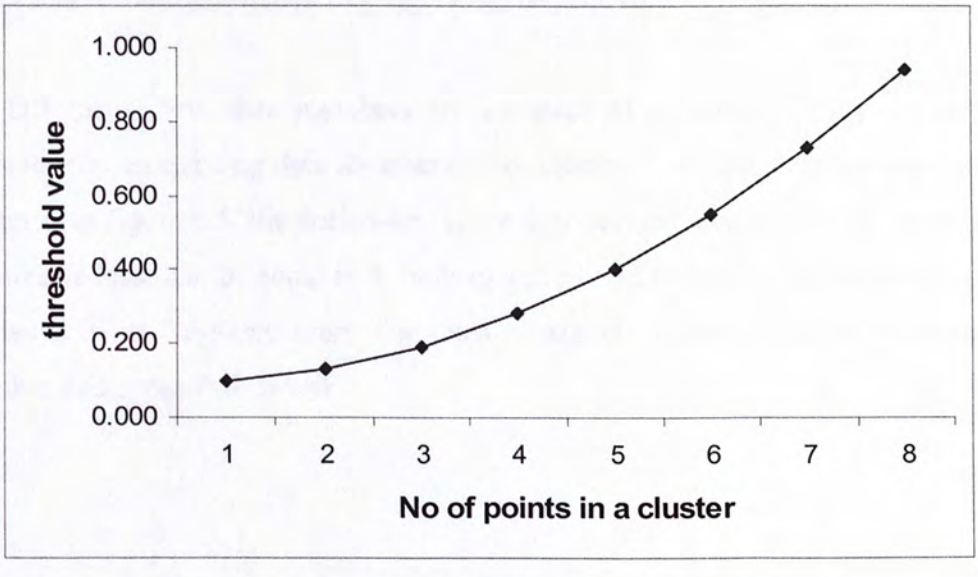

Figure 6.3 Threshold Function with small size

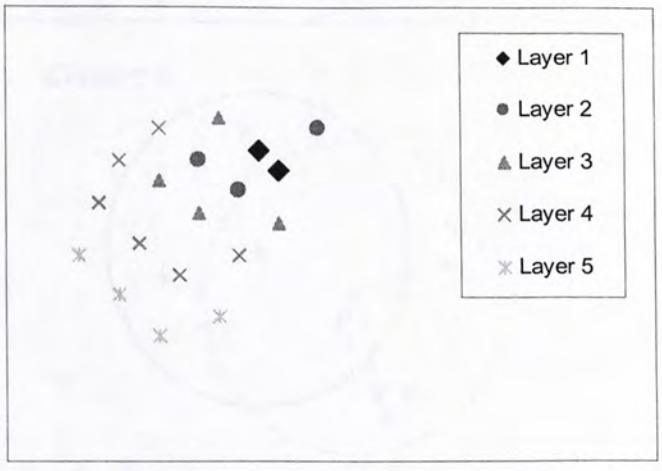

Figure 6.4 Elongated-shape Cluster built with small threshold size

6.4.2 Expansion process

Although the proposed algorithm can provide elongated-shape clusters, the actual expansion process is differed from that of Density-based Scan algorithm. DB Scan actively "invites" the unlabeled points to become a data member of a cluster while our proposed algorithm passively "waits" an unlabeled point to join a cluster as a member. Figure 6.5 and 6.6 show an example of the expansion process for DB Scan and Scalable Distance Clustering algorithm respectively.

For DB Scan, new data members are assigned to a cluster if they are density-reachable by an existing data member of that cluster. Consider the data member p of cluster C in figure 6.5, the dotted-line is the user-defined area, if the minimum points required is less than or equal to 4, both q_1 and q_2 will be labeled as the data member of cluster C at the same time. The cluster expands starting from an existing data member and spreads outward.

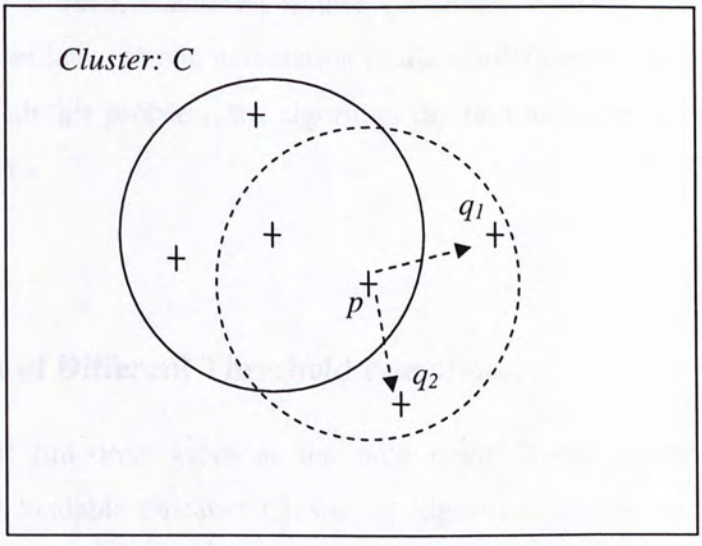

Figure 6.5 Expansion process for Density-based Scan Algorithm

Our proposed algorithm expands a cluster with different approach. Refer to figure 6.6, if the average distance between a non-data member q and *n-nearest neighbors*, i.e. p_1, p_2 and p_3 if n is defined as 3, is within the predefined threshold value obtained from the input threshold function with respect to the current cluster size, the point q becomes a new data member.

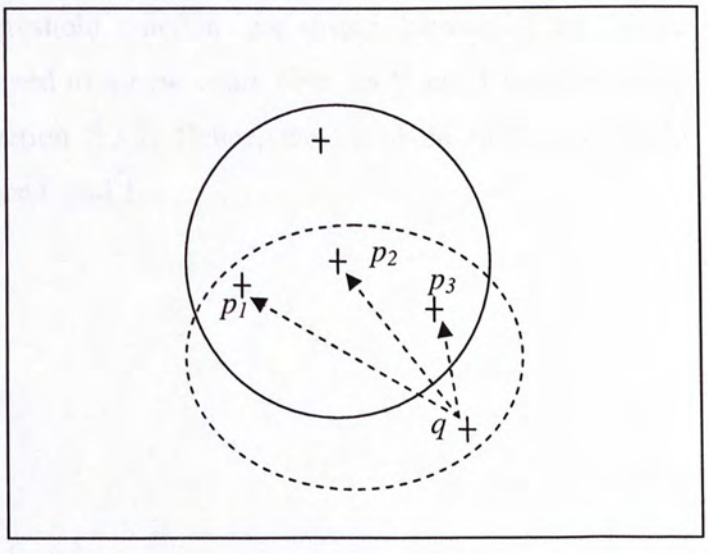

Figure 6.6 Expansion process for Scalable Distance Clustering Algorithm

Compare with DB Scan, Scalable Distance Clustering algorithm can make sure the new data member has a strong association to the whole/part of the cluster, but not a single point. With this property, the algorithm can minimize the rise of merging two unrelated clusters.

6.4.3 Impact of Different Threshold Functions

Since threshold functions serve as the only input, it can greatly influence the performance of Scalable Distance Clustering Algorithm. In this section, the impact of different threshold functions on the formation of clusters is being studied.

In order to expand a cluster, the threshold values is loosen when the cluster size increases, so that points which are related to the same concept with weaker strength can become one of the cluster members. Therefore, we always use strictly increasing function as an input for Scalable Distance Clustering Algorithm.

6.4.3.1 Range of Threshold Functions

Although the threshold functions are strictly increasing, the distance between two terms is normalized to a new value between 0 and 1 using min-max normalization according to section 5.3.1. Hence, the threshold functions should have a nature boundary between 0 and 1.

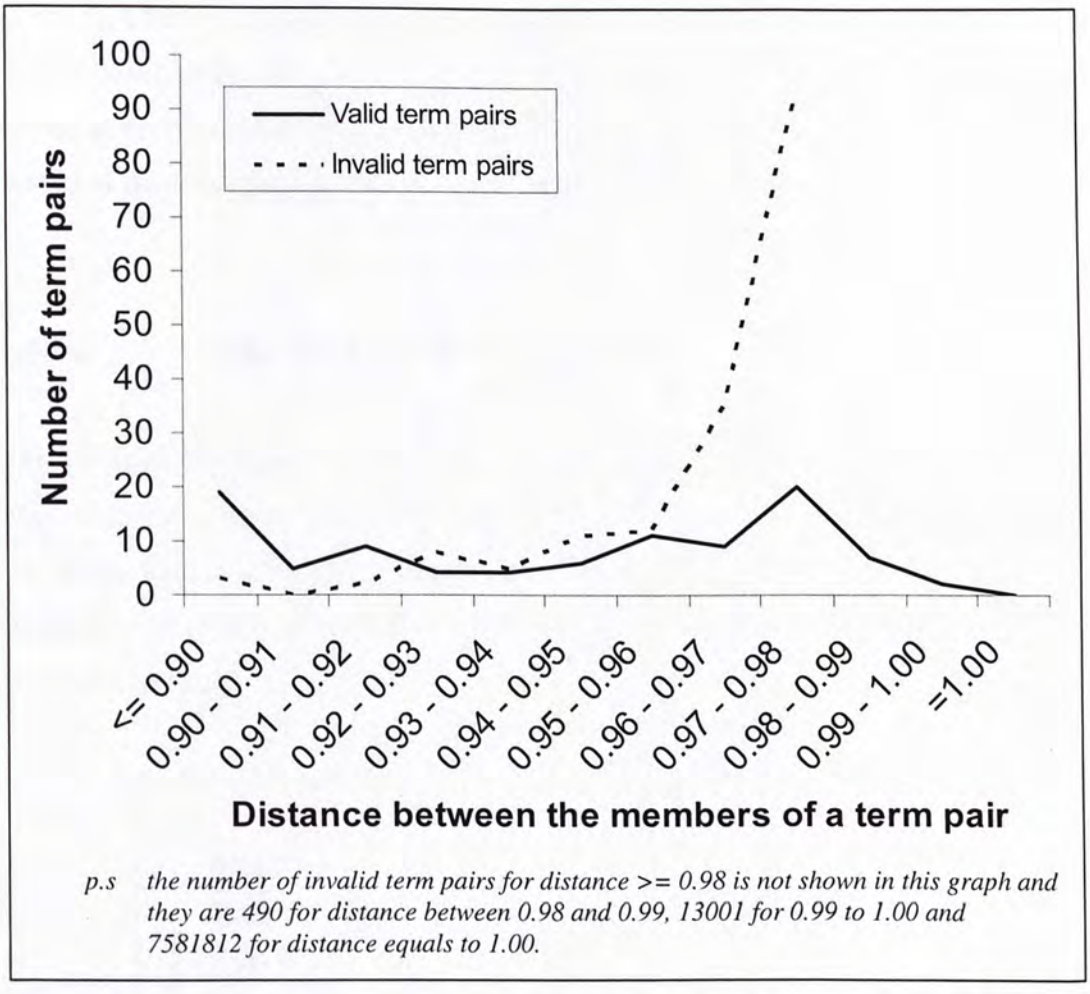

Figure 6.7 Number of term pairs describing the same concept and number of term pairs describing different concepts w.r.t the distance between the terms

Figure 6.7 studies the relationship between the validity and the distance between the members of the term pairs based on the digital camera reviews. A term pair is valid if its members describe the same concept. On the contrary, a term pair is invalid if its members describe different concepts. The number of invalid term pairs increase steadily when the distance between the members of the term pairs increase. In order to reduce the chance of including too many noisy data in a concept cluster, the upper bound of the threshold functions should be less than the natural upper bound.

Since the threshold functions are strictly increasing, the lower bound should be the first member of the functions, i.e. the threshold value for getting the initial seeds,

i_thres, in our proposed algorithm. This initial threshold value is directly related to the number of clusters obtained. Based on the above property, the lower bound can be found using alternative method. We can specify the number of target clusters or initial seeds and determine the corresponding threshold value. Hence, the lower bound of the threshold functions should be greater than the natural lower bound 0.

6.4.3.2 Shape of Threshold Functions

The shape of threshold functions plays an important role in controlling the speed of expansion for a cluster. Figure 6.8 shows three possible shapes for the threshold functions. Figure 6.9 helps to predict the expansion style using the three threshold functions by showing the number of term pairs that are within the corresponding threshold value.

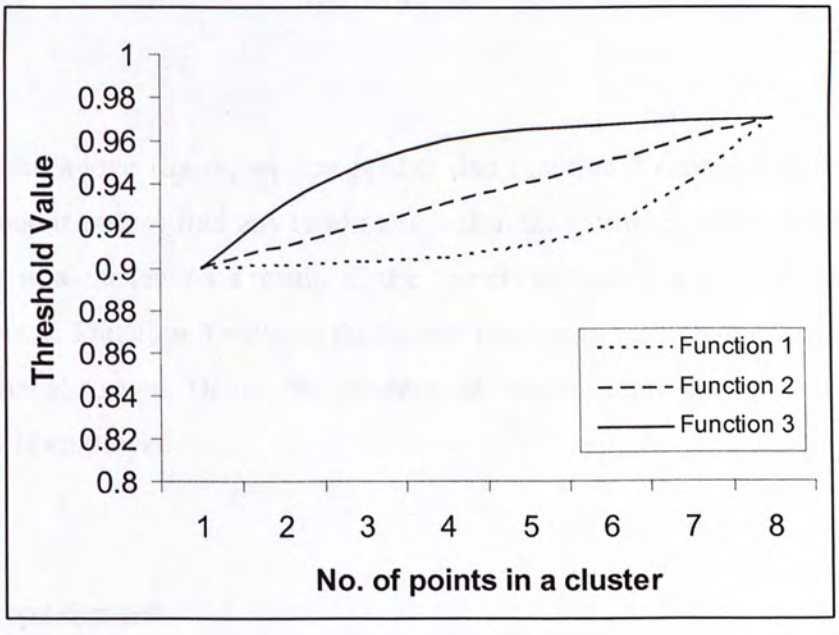

Figure 6.8 Possible shapes for the threshold function

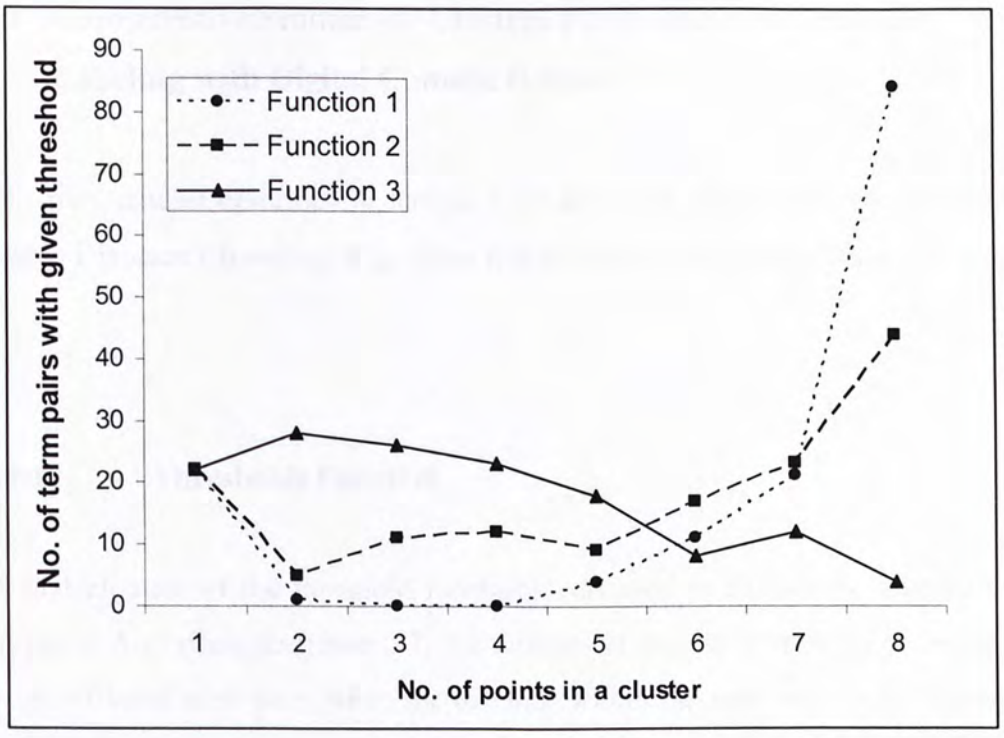

Figure 6.9 Number of term pairs being considered with respect to the cluster size for the three possible functions

Based on the above figure, we can predict that function 1 cannot help the clusters expand since it cannot find any term pairs within the given threshold when there are two points in a cluster. As a result, all the clusters extracted only have two members for function 1. Function 3 extracts the largest number of term pairs when the cluster size is equaled to two. Hence, the clusters can expand more easily when threshold function 3 is employed.

6.5 Experiment

In this section, we extracted product feature keywords clusters by using the proposed Scalable Distance Clustering Algorithm. We evaluated the effectiveness for extracting the product feature sentences by investigating the clusters extracted as well as the accuracy of the sentences' label.

6.5.1 Comparative Studies for Clusters Formation and Sentences Labeling with Digital Camera Dataset

In this part, dataset described in section 3.4.1 was used. The result generated by the Scalable Distance Clustering Algorithm was compared with that obtained in chapter 5.

6.5.1.1 Thresholds Function

The first element of the threshold functions was used to extract the initial sets of seed pairs. According to figure 6.7, the number of invalid term pairs exceeded the number of valid term pairs when the distance within the term pairs was larger than 0.92. In the following experiments, 0.92 was chosen as the first element of the threshold functions. Figure 6.10 shows 5 sample threshold functions which had a wider range of cluster-point size. Since the product feature clusters usually have a few numbers of keywords, they can help to make sure all the points within a cluster would be taken into consideration when the clusters expand.

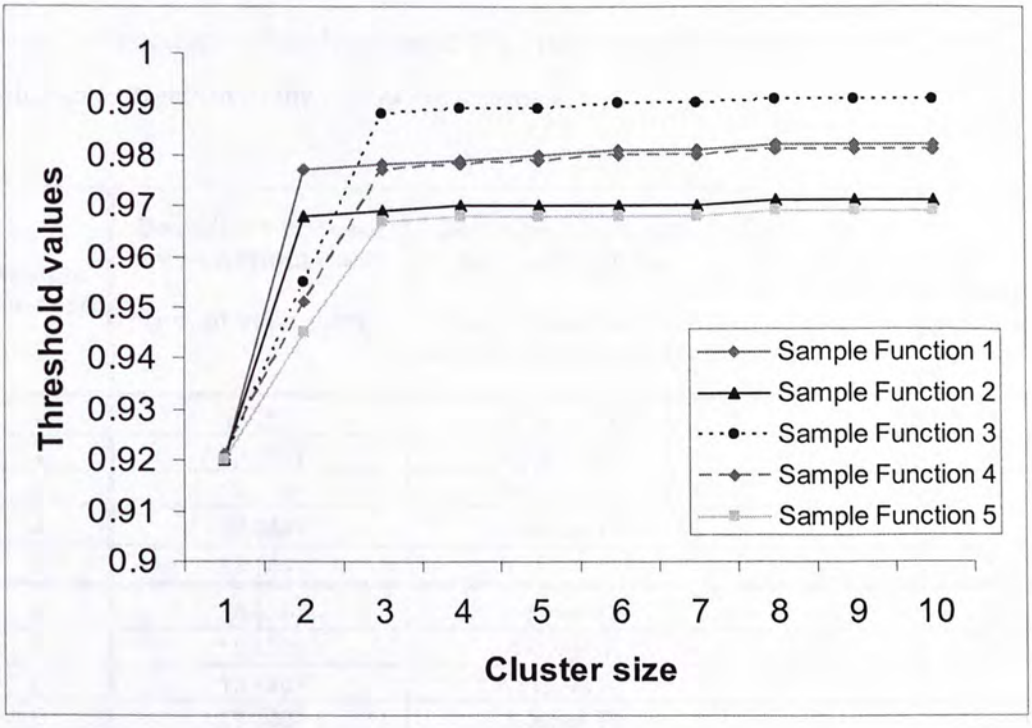

Figure 6.10 Threshold functions with larger size (Sample function 1 – 5)

Figure 6.11 shows another 4 sample threshold functions which had a narrower range of cluster-point size. It can help us to understand the relationship between the threshold functions and the clusters.

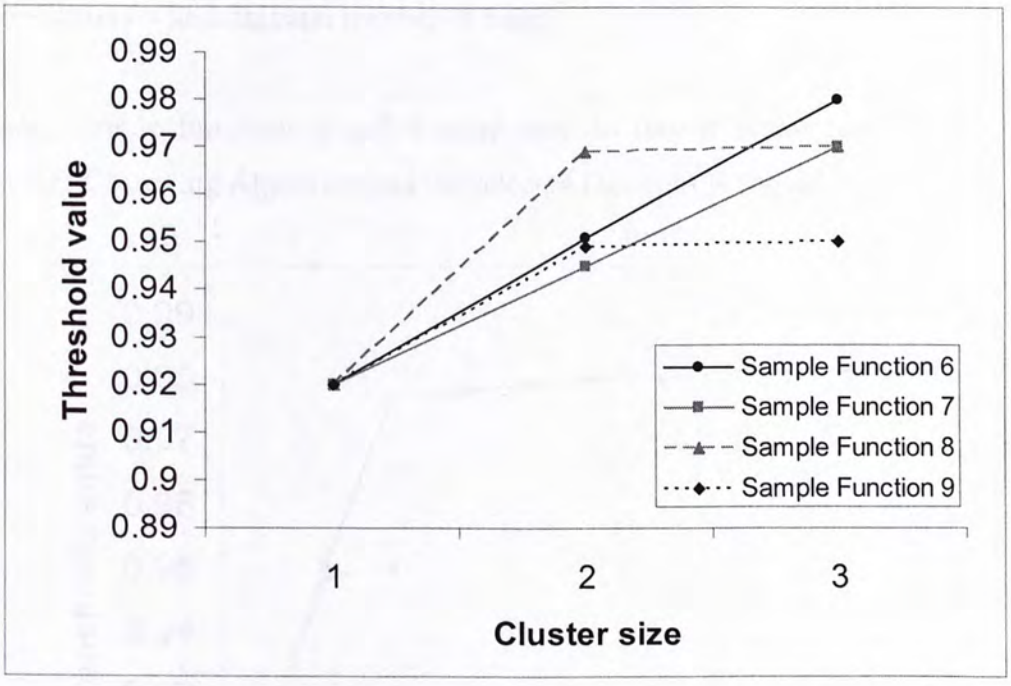

Figure 6.11 Threshold functions with smaller size (Sample function 6 – 9)

Table 6.1 shows the evaluation for the clusters extracted by 9 distinct sample threshold functions. Based on table 6.1, two sample functions were chosen for further investigation in the following sections.

Sample Function	Describe 1 concept only - without noise (no. of valid term)	Describe 1 concept only - with noise (no. of valid terms vs no. of noisy term)	Describe >1 concepts	No. of invalid cluster
1	8 (23)	6 (18 vs 12)	2	3
2	13 (40)	4 (12 vs 5)	0	5
3	12 (30)	1 (6 vs 1)	2	5
4	**16 (44)**	**1 (6 vs 1)**	0	5
5	17 (44)	0	0	5
6	14 (39)	1 (7 vs 4)	1	5
7	16 (39)	1 6 vs 3)	0	5
8	13 (40)	5 (12 vs 8)	0	4
9	**17 (40)**	**1 (6 vs 1)**	0	5

Table 6.1 Clusters Evaluation (Sample function 1 - 9)

Consider sample function 1 – 5, sample functions 4 could extract the largest number of valid terms for the clusters describing a single concept. Although, one of the clusters contained a noisy term, it still gave a better performance compared with the others. For sample functions 6 – 9, sample function 9 extracted the larger number of valid clusters with minimum number of noise.

Hence, sample functions 4 and 9 were used to further investigate the Scalable Distance Clustering Algorithm and the selected functions is shown in figure 6.12.

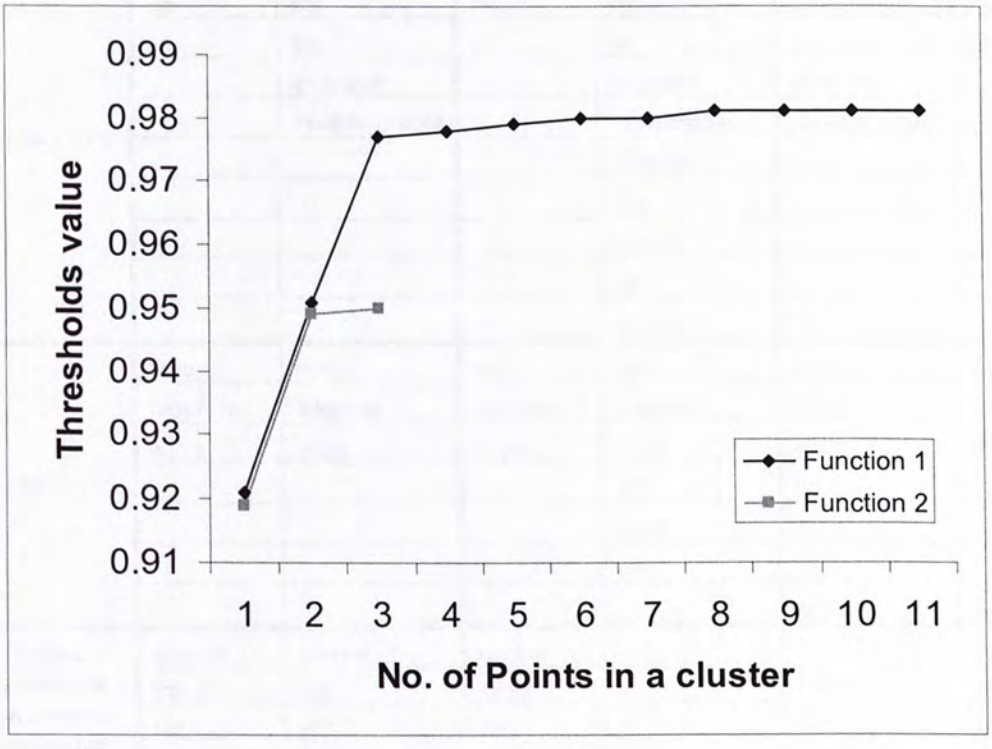

Figure 6.12 Selected Input Thresholds Functions for R_{dc}

6.5.1.2 Clusters Evaluation

In order to illustrate the effectiveness of our proposed algorithm, we compared the product feature clusters extracted by Scalable Distance Clustering Algorithm using the threshold functions described in figure 6.12 with the best results obtained by hierarchical clustering algorithm and DB Scan algorithm. According to the experiments in section 5.3, hierarchical clustering with cutting threshold = 0.91

provided the best macro F-score (72.50%) and the cutting threshold = 0.95 provided the best micro F-score (66.14%) while DB Scan with EPS = 0.93 gave the best macro F-score (70.82%) and the best micro F-score (65.51%). The comparisons of all the product feature clusters are list in Appendix C.

PF Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
	Cut thres = 0.91	Cut thres = 0.95	Eps =0.93	Eps = 0.96	Function 1	Function 2
battery life	battery	battery	battery	battery	battery	battery
	life	life	life	life	life	life
	aa	aa	aa	aa	aa	aa
		charger		***charger***	***charger***	
		rechargeable		***rechargeable***	***rechargeable***	
				charge		
				fps		
				frame		
				per		
				second		
lens	wide	wide	wide	wide	wide	wide
	telephoto	telephoto	telephoto	telephoto	telephoto	angle
	angle	angle	angle	angle	angle	telephoto
				kit	***kit***	
				lens	***lens***	
				mm	***mm***	
					efs	
video frame rate / continuous shooting	second	second	second		second	second
	frame	frame	frame		frame	frame
	per	per	per		per	per
		fps			***fps***	

Table 6.2 Product Feature Clusters (Sample 1)

According to table 6.2, our proposed algorithm can help to expand the product feature clusters more effectively compared with hierarchical clustering with tight cutting threshold and DB Scan with small *eps*. Consider the product features cluster "battery life", hierarchical clustering algorithm with cutting threshold = 0.91 and DB Scan with *eps* = 0.93 could only extract the keywords "battery", "life" and "AA", but our proposed algorithm could also extract other related terms "charger" and "rechargeable".

Although we can obtained the same sets of keywords for "battery life" when we relaxed the cutting threshold for hierarchical clustering algorithm, the number of invalid clusters increased. When the cutting threshold = 0.95, > 40% of the clusters were invalid while < 25% of the clusters were invalid for our proposed algorithm with the given threshold function. DB Scan extracted less invalid clusters, but the clusters could easily merge when the *eps* relaxed. The product feature "battery life" and "continuous shooting or video frame rate" merged into one cluster when *eps* = 0.96.

PF Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
	0.91	0.95	0.93	0.96	Function 1	Function 2
white balance	white	white	white	white	white	white
	balance	balance	balance	balance	balance	balance
		black	***black***	***black***		
LCD screen	lcd	lcd		lcd	lcd	lcd
	screen	screen		screen	screen	screen
		view		***view***		
picture quality (noise and iso)	noise	noise		noise	noise	noise
	iso	iso		iso	iso	iso
		high		***high***		
		higher		***higher***		
		end		***end***		
				anyone		
				recommend		

Table 6.3 Product Feature Clusters (Sample 2)

Table 6.3 further shows that Scalable Distance Clustering Algorithm could avoid some non-informative keywords to be group in the product features clusters. It was obvious words like "high", "end", etc were some common and general terms and they can greatly influence the accuracy of the labeling. For the product feature cluster "picture quality (noise and ISO)", the recall kept constant while the precision changed from 50% to 15% when "high", "higher" and "end" were added to the original cluster with the keywords "noise" and "ISO".

6.5.1.3 Labeling of Sentences

The sets of product feature clusters found in Appendix C were used to label the digital camera sentences. The comparison of the result is shown in the following table.

		Hierarchical Clustering		DB Scan		Scalable Thresholds Clustering Algorithm	
		0.91	0.95	0.93	0.96	Function 1	Function 2
Macro	Recall	87.37%	76.23%	81.23%	86.60%	84.29%	82.60%
	Precision	61.95%	56.40%	62.78%	57.00%	64.38%	64.12%
	F-score	**72.50%**	64.83%	**70.82%**	68.75%	***73.00%***	***72.19%***
Micro	Recall	66.85%	69.09%	59.92%	54.79%	73.11%	72.72%
	Precision	64.72%	63.43%	72.24%	61.32%	72.63%	71.85%
	F-score	65.77%	**66.14%**	**65.51%**	57.87%	***72.87%***	***72.28%***
# of invalid clusters		4	19	***0***	2	5	5
# of valid clusters		14	***25***	8	12	17	17

Table 6.4 Comparison of the accuracy for the sentences labeling (Digital Camera)

For DB Scan, although it extracted the least number of invalid clusters, it provided the worst macro and micro F-score compared with the other two algorithms. For Hierarchical Clustering, the number of valid and invalid clusters increased rapidly when the cutting threshold relaxed. For our proposed Scalable Distance Clustering Algorithm, it could provide the best macro and micro F-score with a single function and it could give a well-balanced between the number of invalid and valid clusters.

6.5.2 Experiments with New Datasets

We collected another three sets of consumer reviews about personal computers, mobile and MP3 from amazon.com to test on the Scalable Distance Clustering Algorithm. The distribution of the reviews is given in the following tables.

Product Type	Notation	# of models	# of reviews	# of sentences
Personal Computers	R_{pc}	10	223	2647
Mobile	R_{mobile}	9	225	2867
MP3	R_{mp3}	6	210	3035

Table 6.5 Distribution of all the datasets

Similar to the analysis described in section 6.4.3, each dataset above was studied to derive two independent thresholds functions. According to the distribution of the distance between all term pairs in a dataset, optimal lower bound and upper bound were found so that sufficient initial seeds were extracted and non-informative terms were unable to join. Also, based on our observations, concave-shaped monotonic increasing function could facilitate the expansion of clusters.

6.5.2.1 Experimental Results for Personal Computer Dataset

Figure 6.13 shows another two thresholds functions derived to control the expansion of clusters for personal computer dataset using the Scalable Distance Clustering Algorithm.

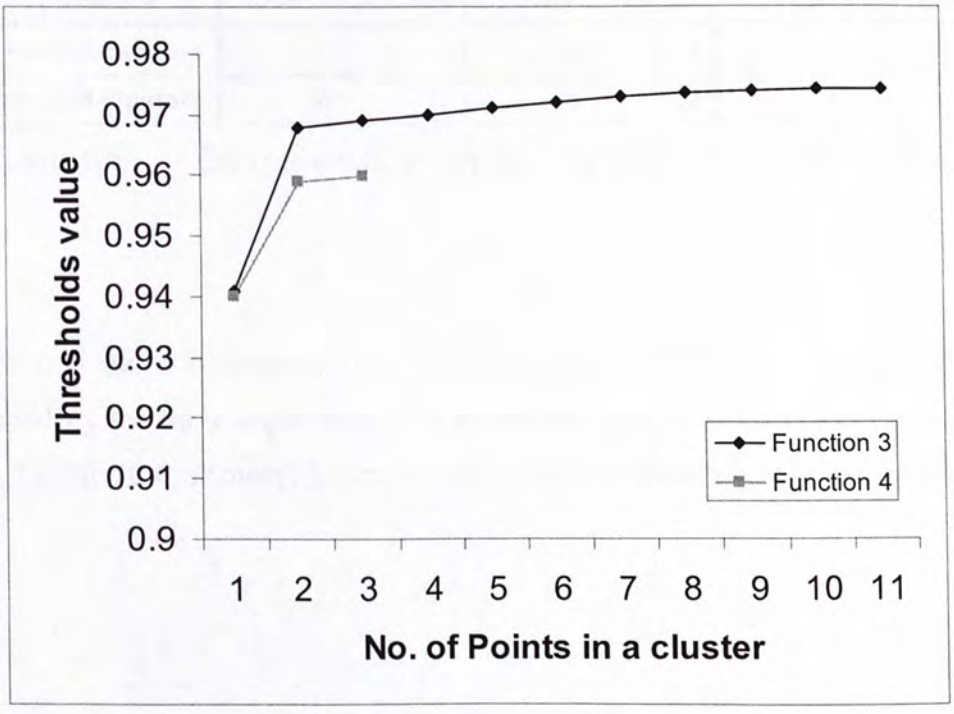

Figure 6.13 Input Thresholds Functions for R_{pc}

The comparison between the three algorithms is listed in table 6.6. The details of the concept clusters extracted is shown in Appendix D. By setting the cutting threshold = 0.93 and 0.94 for the Single-linkage Hierarchical Clustering algorithm could obtain the best macro and micro f-score respectively. While DB Scan algorithm with EPS = 0.93 gave the best macro f-score (72.87%) among the three algorithms, but the micro F-score was comparative low since it could only extract 9 valid clusters and hence less sentences were being tagged. Similar to the case of extracting product feature sentence from digital camera reviews, our proposed algorithm could obtain a relatively good macro and micro F-score with a single function.

		Hierarchical Clustering		DB Scan		Scalable Thresholds Clustering Algorithm	
		0.93	0.94	0.93	0.96	Function 3	Function 4
Macro	Recall	*76.52%*	*73.64%*	*72.12%*	*77.76%*	*74.77%*	*73.31%*
	Precision	*66.51%*	*66.75%*	*73.64%*	*62.40%*	*68.02%*	*65.89%*
	F-score	***71.16%***	*70.03%*	***72.87%***	*69.24%*	***71.23%***	*69.40%*
Micro	Recall	*71.60%*	*73.95%*	*66.80%*	*69.97%*	*74.12%*	*73.71%*
	Precision	*73.74%*	*75.47%*	*70.45%*	*71.48%*	*76.98%*	*74.76%*
	F-score	*72.66%*	***74.70%***	*68.58%*	***70.72%***	***75.52%***	*74.23%*
# of invalid clusters		6	7	***1***	4	6	6
# of valid clusters		16	17	9	12	17	***19***

Table 6.6 Comparison of the accuracy for the sentences labeling (R_{pc})

Tables 6.7 and 6.8 compare the concept clusters "CPU" and "Memory (Ram)" extracted by the three algorithms. The keywords "ghz" was related to the speed of CPU, but not the memory, hence it wass actually a member of the concept about CPU.

Concepts	Scalable Thresholds Clustering	Hierarchical Clustering				
	Function 1	0.92	0.93	0.94	0.95	0.96
CPU	duo	duo	duo	duo	duo	duo
	core	core	core	core	core	core
	dual	dual	dual	dual	dual	dual
	processor	processor	processor	processor	processor	processor
	intel	intel	intel	intel	intel	intel
						faster
	ghz					
Memory (Ram)	gb	gb	gb	gb	gb	gb
	ram	ram	ram	ram	ram	ram
	memory			memory	memory	memory
	mb	mb	mb	mb	mb	mb
	upgraded		upgraded	upgraded	upgraded	upgraded
	gig					gig
				ghz	***ghz***	***ghz***

Table 6.7 Comparison between two concept clusters formed by Scalable Distance Clustering and Hierarchical Clustering

Concepts	Scalable Thresholds Clustering	DB Scan				
	Function 1	0.93	0.94	0.95	0.96	0.97
CPU	duo	duo	duo	duo	duo	
	core	core	core	core	core	
	dual	dual	dual	dual	dual	
	processor	processor	processor	processor	processor	
	intel	intel	intel	intel	intel	
					faster	
	ghz					
Memory (Ram)	gb	gb	gb	gb	gb	gb
	ram	ram	ram	ram	ram	ram
	memory		memory	memory	memory	memory
	mb	mb	mb	mb	mb	mb
	upgraded	upgraded	upgraded	upgraded	upgraded	upgraded
	gig				gig	gig
			ghz	***ghz***	***ghz***	***ghz***
						duo
						core
						dual
						processor
						intel
						faster
						based
						extreme

Table 6.8 Comparison between two concept clusters formed by Scalable Distance Clustering and DB Scan

The two tables clearly showed that both hierarchical clustering and DB Scan failed to cluster the terms with the concept about CPU. For DB Scan, the two distinct concepts even merged when the EPS was loosen. For our proposed Scalable Distance Clustering Algorithm, it could place the term "ghz" in the concept cluster about "CPU". Although the algorithm was able to form concept cluster with better quality, the improvement was not significant when the terms were used to identify the product feature sentences and this may due to the lack of sample (there are only 150 out of 2647 sentences commented on "CPU").

6.5.2.2 Experimental Results for Mobile Dataset

According to the criteria for the deviation of thresholds function, two thresholds functions shown in figure 6.14 were derived based on the distribution of mobile dataset.

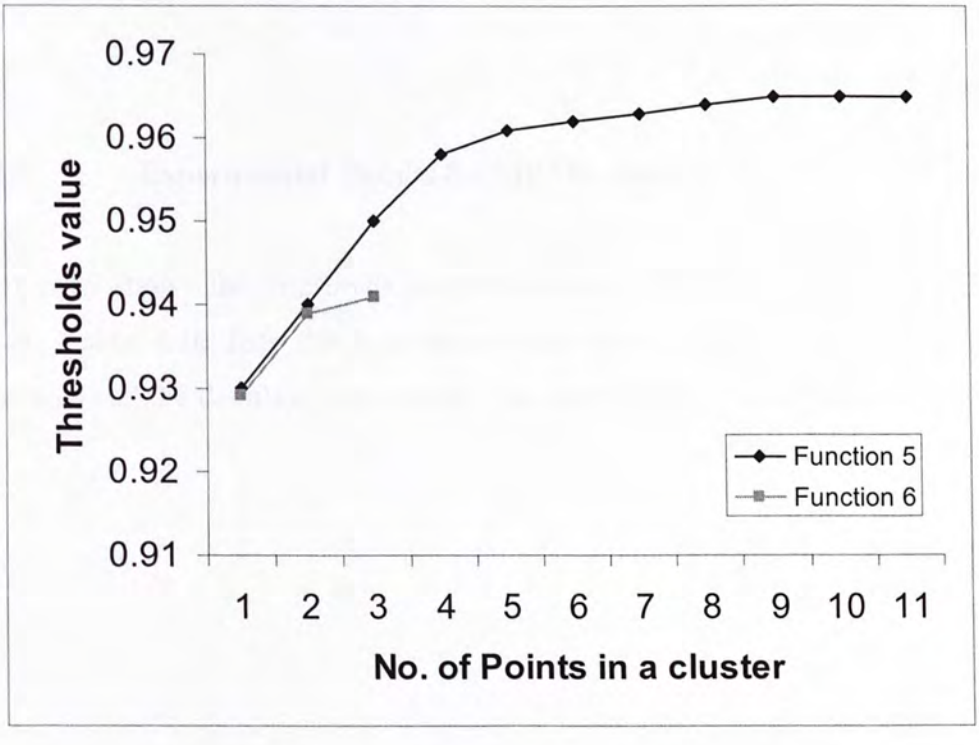

Figure 6.14 Input Thresholds Functions for R_{mobile}

Table 6.9 lists the comparison between the best results obtained by hierarchical clustering, DB Scan and Scalable Distance Clustering respectively. The details of the concept clusters extracted are shown in Appendix E.

		Hierarchical Clustering		DB Scan		Scalable Thresholds Clustering Algorithm	
		0.93	0.94	0.92	0.94	Function 5	Function 6
Macro	Recall	*81.78%*	*82.65%*	*84.15%*	*84.15%*	*80.68%*	*80.68%*
	Precision	*71.41%*	*71.01%*	*70.71%*	*69.79%*	*72.67%*	*72.67%*
	F-score	*76.25%*	*76.39%*	***76.85%***	*76.30%*	*76.47%*	*76.47%*
Micro	Recall	*74.91%*	*74.91%*	*58.71%*	*61.71%*	*75.28%*	*75.28%*
	Precision	*75.00%*	*74.75%*	*65.25%*	*67.10%*	*77.33%*	*77.33%*
	F-score	*74.95%*	*74.83%*	*61.81%*	*64.29%*	***76.29%***	***76.29%***
# of invalid clusters		8	10	***0***	1	8	8
# of valid clusters		***24***	***24***	6	8	***24***	***24***

Table 6.9 Comparison of the accuracy for the sentences labeling (R_{mobile})

6.5.2.3 Experimental Results for MP3 Dataset

Figure 6.15 shows the thresholds functions derived from the distribution of MP3 dataset. Table 6.10 lists the best macro and micro f-score found by the three algorithms and the details of the concept clusters extracted are shown in Appendix F.

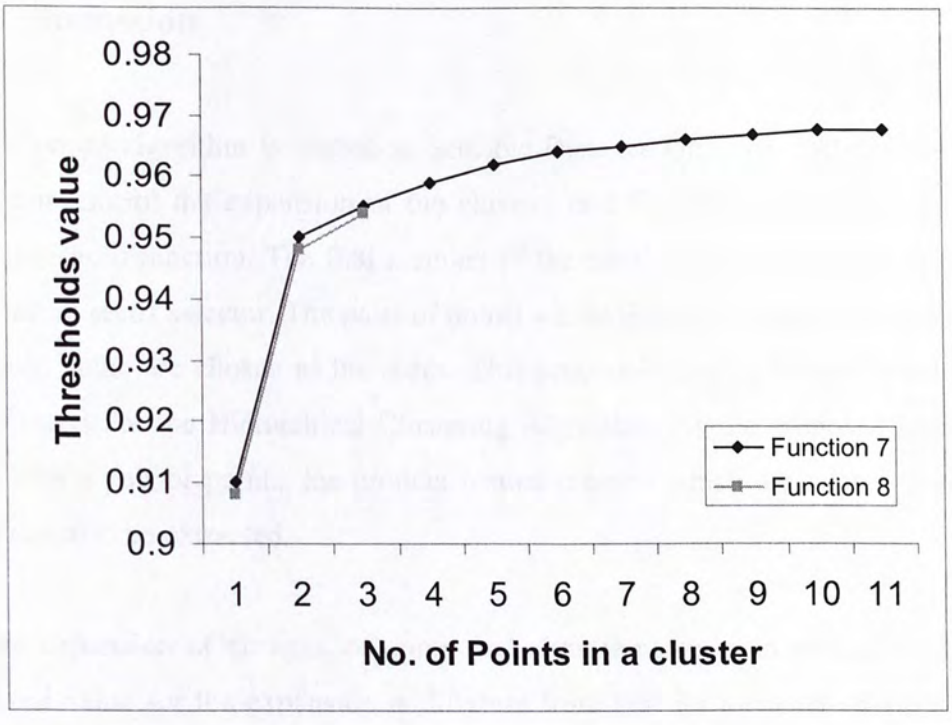

Figure 6.15 Input Thresholds Functions for R_{mp3}

		Hierarchical Clustering		DB Scan		Scalable Thresholds Clustering Algorithm	
		0.92	0.95	0.95	0.96	Function 5	Function 6
Macro	Recall	*89.62%*	*86.14%*	*94.15%*	*92.57%*	*89.94%*	*89.94%*
	Precision	*66.08%*	*64.40%*	*57.92%*	*58.62%*	*65.31%*	*65.31%*
	F-score	***76.07%***	*73.70%*	*71.72%*	*71.79%*	*75.67%*	*75.67%*
Micro	Recall	*65.21%*	*68.01%*	*67.65%*	*68.53%*	*74.88%*	*74.88%*
	Precision	*75.92%*	*75.89%*	*75.34%*	*78.57%*	*76.06%*	*76.06%*
	F-score	*70.16%*	*71.74%*	*71.29%*	*73.21%*	***75.47%***	***75.47%***
# of invalid clusters		9	18	***4***	***4***	6	6
# of valid clusters		15	***18***	9	13	15	15

Table 6.10 Comparison of the accuracy for the sentences labeling (R_{mp3})

6.6 Discussion

Our proposed algorithm is named as Scalable Distance Clustering Algorithm since users can control the expansion of the clusters in a flexible way by adjusting the input threshold function. The first member of the input threshold function is served as an initial seeds selector. The pairs of points whose distance is shorter than the first threshold value are chosen as the seeds. This process is similar to the formation of new clusters for the Hierarchical Clustering Algorithm. As the proposed algorithm starts with a pair of points, the product feature clusters which are related to only 2 terms can also be extracted.

For the expansion of clusters, our proposed algorithm works in another way. The threshold value for the expansion is different from that for the seed selection. It is adjusted according to the size of the clusters and the threshold function. By splitting the algorithm into the above steps, it can make sure the clusters can be expanded effectively with relaxed threshold value while the clusters with loosen connection are not formed at the same time. Since those clusters with loosen connection are usually non-informative, the Scalable Threshold Clustering Algorithm can help to maximize the number of useful terms and minimize the number of non-informative clusters at the same time. Although some of the clusters extracted may still not be related to any product features, the relaxation of the number of terms within a cluster is well worth. The number of valid clusters extracted can overcome the weakness according to our experiment shown section 6.5.

The flexible threshold function can also help to adjust the expansion style of the clusters. By using threshold function with large range, all the points within the cluster take into consideration when a point is added. Hence, the point must be closely related to all the member of the cluster so that the point can be added successfully. For threshold function with narrow range, only parts of the points within the cluster are considered. It is easier for a cluster to expand, but elongated-shape clusters can be obtained and affect the accuracy as a result. Based on our experiment and understanding, it is more suitable to use wide range threshold function for the extraction of product feature clusters.

To conclude, the extraction of product feature sentences by using our proposed algorithm provides the best micro and macro F-score compare with Hierarchical Clustering Algorithm and Density-Based Scan Algorithm, but there is still room to improve the labeling of product feature sentences.

7. Conclusion and Future Work

The World Wide Web gathers information and users' opinions from all over the world with the help of various consumers' websites, discussion groups and forums. Comments from the consumers' websites such as Amazon.com allow users to compare the strengths and weaknesses of different products and make a wise choice during purchasing. Producers can also benefit from such comments by analyzing the consumers' concern, so that they can further improve their products. Nowadays, searching and analyzing others' opinion from the web become a common practice for most users.

Unfortunately, the amount of available information greatly exceeds the users' needs. The hottest the topics, the much information can be found. Therefore, it is very time consuming if we analyze them with human effort. Furthermore, the opinions may contain many unwanted information and some of them may even not well-layout, making the retrieve of users' interested information more difficult.

In order to facilitate the search of users' interested information with minimum human effort, many researchers developed different technique to categorize the online users' opinion according to the polarity of the opinion or features being described in the opinion. With the help of such technique, tailor-made information can be obtained for each individual.

7.1 Compare with Existing Work

Some previous researches focus on classifying a document, but based on our observation, a review usually comments on more than one product features. Hence, it is better to analysis each sentences rather than the review as a whole. Since a sentence contains a few numbers of words only, it may not be able to provide sufficient information to capture the idea being described in a single sentence. Also, some sentences may not associate to any product features. According to our

collected data, less than 50% of sentences are truly describing the product features. These are the main challenges being faced in sentence-based analysis.

In addition, most of the previous researches employ the Natural Language Processing (NLP) approach to general language pattern based on part-of-speech (POS) and grammar rules. Despite the fact that most online opinion are written informally without spelling or grammar check and the users may use words that are not found in the dictionary during writing, it greatly influences the accuracy of the NLP tagging. According to Liu et. al. (2005), only 52% of the data can be correctly tagged. In this work, we would like to focus on using alternative approach in extracting product feature sentences.

Despite the failure of the existing methods, we used an alternative approach to solve the problem. In order to minimize the effect on incorrect label of NLP tagging, we conducted experiments using supervised learning methods to generate classifiers without considering the POS of the terms. Although both Class Association Rules and the Naïve Bayesian Classifier provide a promising result with around 0.7 for the F-score, it requires human effort to label the training data and only the predefined classes within the training data can be obtained. In order to overcome the weakness being faced by the supervised learning methods, we employ the unsupervised technique to group the product feature sentences in chapter 4. Due to the large portion of non-product feature sentences and the limit of information, e.g. the co-occurrence of terms between two sentences, this approach cannot work effectively.

In chapter 5, we introduced the idea of concept clustering and show that it works fine on the extraction of product feature sentences. Concept clustering refers to the technique in organizing the terms that can be used to describe the same idea (also known as concept) into concept cluster. Similarities between terms are calculated based on the co-occurrence frequency in the input set of sentences. We further improved the accuracy of labeling with the help of our proposed algorithm named "Scalable Distance Clustering Algorithm". Our experiments showed that it worked effectively in the labeling of consumer review sentences.

7.2 Contribution & Implication of this Work

This work makes two main contributions to the extraction of web opinion on consumer reviews. Firstly, it applied the idea of concept clustering to generate classifiers automatically. Secondly, it proposed a new unsupervised learning technique called "Scalable Threshold Clustering Algorithm" to facilitate the extraction of clusters with high flexibility.

The proposed algorithm is called as Scalable Threshold Clustering Algorithm since the users can control the expansion of the clusters in a flexible way by adjusting a user-defined threshold function. The threshold function is acted as the input to determine the flexible threshold value with respect to the size of clusters when clusters expand. The algorithm is able to extract product feature clusters which are related to only 2 terms. Since clusters with closer distance are usually more informational, by setting a tight initial threshold, we can limit the number of clusters formed and their quality. After initialization, the threshold value can gradually loosen with respect to the cluster size so that other related terms, which have weak association power to the corresponding concepts, can join the clusters as a member. With the help of the threshold functions, the Scalable Threshold Clustering Algorithm can maximize the number of useful terms and minimize the number of non-informative clusters at the same time. It can also control the expansion style of the clusters by adjusting the range of the function. For threshold functions with wide range, a point should be closed to all members within the cluster in order to add it into the cluster. If a threshold function has narrow range, only parts of the points are considered. Threshold function with wide range works better in the extraction of product feature clusters based on our experiment.

It is clearly shown that concept clustering can help to extract product feature with minimum human effort. Compare with Hierarchical Clustering Algorithm and Density-Based Scan Algorithm, our proposed algorithm is able to extract higher quality concept clusters and give a better combination of macro and micro f-score in the extraction of product feature sentences for digital camera, personal computer, mobile and MP3 reviews.

7.3 Future Work & Improvement

In the future, we can try with several streams of follow-up studies. Firstly, as the labeling of datasets and the evaluation of clusters were done by only one person, biases may be introduced. Therefore, it would be better if more people involve in the labeling and independent evaluators are used to determine the performance of the clustering algorithms. Secondly, the threshold function for our proposed algorithm is only generated semi-automatically in current design, which is first generated based on the distribution of the sets of sentences and then fine-tune and evaluate according to the observation of users. As future work, we can further study the relationship between the threshold functions and the statistical data so that they can be generated automatically.

We can also test if concept clustering or the proposed Scalable Distance Clustering Algorithm can work on the extraction of feature sentences from different domain such as movie reviews and blog post. In addition, the proposed algorithm extracts some non-product feature clusters, which are usually made up of some general word phrases. We can drill down to this problem and develop techniques to remove those invalid clusters automatically.

Lastly, we can study the possibility of implementing the proposed algorithm in actual consumer review websites. User can select their interested type of products and the set of product features extracted by our proposed algorithm will be listed. The sentences with selected product features will be displayed in prior. With the help of this proposed system, users can save time and have a better comparison across different brands of a product.

REFFERENCE

Liu, B., Hu, M., and Cheng, J. 2005. Opinion observer: analyzing and comparing opinions on the Web. In *Proceedings of the 14th international Conference on World Wide Web* (Chiba, Japan, May 10 - 14, 2005). WWW '05. ACM, New York, NY, 342-351.

Jindal, N. and Liu, B. 2006. Identifying comparative sentences in text documents. In *Proceedings of the 29th Annual international ACM SIGIR Conference on Research and Development in information Retrieval* (Seattle, Washington, USA, August 06 - 11, 2006). SIGIR '06. ACM, New York, NY, 244-251.

Liu, J., Wu, G., and Yao, J. 2006. Opinion Searching in Multi-Product Reviews. In *Proceedings of the Sixth IEEE international Conference on Computer and information Technology* (September 20 - 22, 2006). CIT. IEEE Computer Society, Washington, DC, 25.

Pang, B. and Lee, L. 2004. A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting on Association For Computational Linguistics* (Barcelona, Spain, July 21 - 26, 2004). Annual Meeting of the ACL. Association for Computational Linguistics, Morristown, NJ, 271

Cui, H., Mittal, V. and Datar, M. 2006. Comparative Experiments on Sentiment Classification for Online Product Reviews. In *Proceedings of the 21st National Conference on Artificial Intelligence* (Boston, Massachusetts, July 16 - 20, 2006).

Yi, J., Nasukawa, T., Bunescu, R., and Niblack, W. 2003. Sentiment Analyzer: Extracting Sentiments about a Given Topic using Natural Language Processing Techniques. In *Proceedings of the Third IEEE international Conference on Data Mining* (November 19 - 22, 2003). ICDM. IEEE Computer Society, Washington, DC, 427.

Pang, B., Lee, L., and Vaithyanathan, S. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the Acl-02 Conference on Empirical Methods in Natural Language Processing - Volume 10* Annual Meeting of the ACL. Association for Computational Linguistics, Morristown, NJ, 79-86.

Popescu, A. and Etzioni, O. 2005. Extracting product features and opinions from reviews. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing* (Vancouver, British Columbia, Canada, October 06 - 08, 2005). Human Language Technology Conference. Association for Computational Linguistics, Morristown, NJ, 339-346.

Riloff, E. and Wiebe, J. 2003. Learning extraction patterns for subjective expressions. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing - Volume 10* Theoretical Issues In Natural Language Processing. Association for Computational Linguistics, Morristown, NJ, 105-112.

Zhuang, L., Jing, F., and Zhu, X. 2006. Movie review mining and summarization. In *Proceedings of the 15th ACM international Conference on information and Knowledge Management* (Arlington, Virginia, USA, November 06 - 11, 2006). CIKM '06. ACM, New York, NY, 43-50.

Choi, Y., Breck, E. and Cardie, C. 2006. Joint extraction of entities and relations for opinion recognition. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, (Sydney, Australia, July 2006). Association for Computational Linguistics, 431 – 439.

Hu, M. and Liu, B. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD international Conference on Knowledge Discovery and Data Mining* (Seattle, WA, USA, August 22 - 25, 2004). KDD '04. ACM, New York, NY, 168-177.

Ku, L., Liang, Y. and Chen, H. 2006. Opinion extraction, summarization and tracking in news and blog corpora. In *Proceedings of AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs*, AAAI Technical Report SS-06-03, Stanford University, California, US, 2006

Dave, K., Lawrence, S., and Pennock, D. M. 2003. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international Conference on World Wide Web* (Budapest, Hungary, May 20 - 24, 2003). WWW '03. ACM, New York, NY, 519-528.

Ghani, R., Probst, K., Liu, Y., Krema, M., and Fano, A. 2006. Text mining for product attribute extraction. *SIGKDD Explor. Newsl.* 8, 1 (Jun. 2006), 41-48.

Zhang, Z. and Varadarajan, B. 2006. Utility scoring of product reviews. In *Proceedings of the 15th ACM international Conference on information and Knowledge Management* (Arlington, Virginia, USA, November 06 - 11, 2006). CIKM '06. ACM, New York, NY, 51-57.

Agrawal, R., Imieliński, T., and Swami, A. 1993. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international Conference on Management of Data* (Washington, D.C., United States, May 25 - 28, 1993). P. Buneman and S. Jajodia, Eds. SIGMOD '93. ACM, New York, NY, 207-216.

Yang, Y. and Pedersen, J. O. 1997. A Comparative Study on Feature Selection in Text Categorization. In *Proceedings of the Fourteenth international Conference on Machine Learning* (July 08 - 12, 1997). D. H. Fisher, Ed. Morgan Kaufmann Publishers, San Francisco, CA, 412-420.

Li, K. W. and Yang, C. C. 2005. Automatic Crosslingual Thesaurus Generated From the Hong Kong SAR Police Department Web Corpus for Crime Analysis. Research Articles. J. Am. Soc. *Inf. Sci. Technol.* 56, 3 *'05*.

Pang, B. and Lee, L. 2005. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association For Computational Linguistics* (Ann Arbor, Michigan, June 25 - 30, 2005). Annual Meeting of the ACL. Association for Computational Linguistics, Morristown, NJ, 115-124.

Turney, P. D. 2001. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association For Computational Linguistics* (Philadelphia, Pennsylvania, July 07 - 12, 2002). Annual Meeting of the ACL. Association for Computational Linguistics, Morristown, NJ, 417-424.

Hu, M. and Liu, B. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD international Conference on Knowledge Discovery and Data Mining* (Seattle, WA, USA, August 22 - 25, 2004). KDD '04. ACM, New York, NY, 168-177.

Mei, Q., Ling, X., Wondra, M., Su, H., and Zhai, C. 2007. Topic sentiment mixture: modeling facets and opinions in weblogs. In *Proceedings of the 16th international Conference on World Wide Web* (Banff, Alberta, Canada, May 08 - 12, 2007). WWW '07. ACM, New York, NY, 171-180.

Agrawal, R. and Srikant, R. 1994. Fast Algorithms for Mining Association Rules in Large Databases. In *Proceedings of the 20th international Conference on Very Large Data Bases* (September 12 - 15, 1994). J. B. Bocca, M. Jarke, and C. Zaniolo, Eds. Very Large Data Bases. Morgan Kaufmann Publishers, San Francisco, CA, 487-499.

Scaffidi, C., Bierhoff, K., Chang, E., Felker, M., Ng, H., and Jin, C. 2007. Red Opal: product-feature scoring from reviews. In *Proceedings of the 8th ACM Conference on Electronic Commerce* (San Diego, California, USA, June 11 - 15, 2007). EC '07. ACM, New York, NY, 182-191.

Hatzivassiloglou, V. and McKeown, K. R. 1997. Predicting the semantic orientation of adjectives. In *Proceedings of the Eighth Conference on European Chapter of the Association For Computational Linguistics* (Madrid, Spain, July 07 - 12, 1997). European Chapter Meeting of the ACL. Association for Computational Linguistics, Morristown, NJ, 174-181.

Mishne. 2005. Experiments with Mood Classification in Blog Posts. In *Style2005 – the 1st Workshop on Stylistic Analysis of Text for Information Access*, at *SIGIR 2005*, August 2005.

Jindal, N. and Liu, B. 2008. Opinion spam and analysis. In *Proceedings of the international Conference on Web Search and Web Data Mining* (Palo Alto, California, USA, February 11 - 12, 2008). WSDM '08. ACM, New York, NY, 219-230.

Ding, X., Liu, B., and Yu, P. S. 2008. A holistic lexicon-based approach to opinion mining. In *Proceedings of the international Conference on Web Search and Web Data Mining* (Palo Alto, California, USA, February 11 - 12, 2008). WSDM '08. ACM, New York, NY, 231-240.

Archak, N., Ghose, A., and Ipeirotis, P. G. 2007. Show me the money!: deriving the pricing power of product features by mining consumer reviews. In *Proceedings of the 13th ACM SIGKDD international Conference on Knowledge Discovery and Data Mining* (San Jose, California, USA, August 12 - 15, 2007). KDD '07. ACM, New York, NY, 56-65.

Ghose, A. and Ipeirotis, P. G. 2007. Designing novel review ranking systems: predicting the usefulness and impact of reviews. In *Proceedings of the Ninth international Conference on Electronic Commerce* (Minneapolis, MN, USA, August 19 - 22, 2007). ICEC '07, vol. 258. ACM, New York, NY, 303-310.

Yu, H. and Hatzivassiloglou, V. 2003. Towards answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing - Volume 10* Theoretical Issues In Natural Language Processing. Association for Computational Linguistics, Morristown, NJ, 129-136.

Choi, Y., Cardie, C., Riloff, E., and Patwardhan, S. 2005. Identifying sources of opinions with conditional random fields and extraction patterns. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing* (Vancouver, British Columbia, Canada, October 06 - 08, 2005). Human Language Technology Conference. Association for Computational Linguistics, Morristown, NJ, 355-362.

Appendix A. Concept Clustering for DC data with DB Scan (Terms in Concept Clusters)

	Product Feature Description	0.92	0.93	0.94	0.95	0.96	0.97
1	*battery life*	battery	battery	battery	battery	battery	battery
		life	life	life	life	life	life
		aa	aa	aa	aa	aa	aa
				charger	charger	charger	charger
					rechargeable	rechargeable	rechargeable
						charge	charge
						fps	fps
						frame	frame
						per	per
						second	second
2	*size*	fit	fit	fit	fit	fit	fit
		pocket	pocket	pocket	pocket	pocket	pocket
		purse	purse	purse	purse	purse	purse
						bag	bag
						cam	cam
							cybershot
							moment
							simple
							sony
3	*video frame rate / continous shooting*	frame	frame	frame	frame		
		per	per	per	per		
		second	second	second	second		
					fps		
4	*memory storage*	card	card	card	card	card	card
		cf	cf	cf	cf	cf	cf
		gb	gb	gb	gb	gb	gb
		mb	mb	mb	mb	mb	mb
		memory	memory	memory	memory	memory	memory
		reader	reader	reader	reader	reader	reader
		stick	stick	stick	stick	stick	stick
				computer	computer	computer	computer
						download	download
						sd	sd
							getting
							larger
							plan
							door
							free

(To be continued)

	Product Feature Description	0.92	0.93	0.94	0.95	0.96	0.97
5	*lens*	angle	angle	angle	angle	angle	angle
		telephoto	telephoto	telephoto	telephoto	telephoto	telephoto
		wide	wide	wide	wide	wide	wide
					kit	kit	kit
					lens	lens	lens
					mm	mm	mm
							af
6	*shutter speed / aperture*	aperture	aperture	aperture	aperture	aperture	aperture
		lag	lag	lag	lag	lag	lag
		shutter	shutter	shutter	shutter	shutter	shutter
		speed	speed	speed	speed	speed	speed
						background	background
							longer
7	*kit lens*		kit	kit			
			lens	lens			
			mm	mm			
8	*white balance*		balance	balance	balance	balance	balance
			black	black	black	black	black
			white	white	white	white	white
9	*dust on sensor*			clean	clean	clean	clean
				dust	dust	dust	dust
				sensor	sensor	sensor	sensor
10	*model name*				digital	digital	digital
					eos	eos	eos
					rebel	rebel	rebel
					slr	slr	slr
					xt	xt	xt
						camera	camera
11	*lcd screen*				lcd	lcd	lcd
					screen	screen	screen
					view	view	view
12	*picture quality (noise and iso)*				end	end	end
					high	high	high
					higher	higher	higher
					iso	iso	iso
					noise	noise	noise
						anyone	anyone
						recommend	recommend
13	*video*					movie	movie
						sound	sound
						video	video
							auto
							mode
14	*N/A*					amateur	amateur
						photographer	photographer
						professional	professional

(To be continued)

	Product Feature Description	0.92	0.93	0.94	0.95	0.96	0.97
15	*picture quality (color)*					clarity	clarity
						color	color
						detail	detail
16	*N/A*						ago
							had
							ive
							month
							now
							owned
							year
17	*camera performance under low light*						condition
							light
							low
							under
18	*N/A*						pick
							start
							up
19	*ease of use*						easy
							menu
							system
							use
20	*N/A*						manual
							read
							review
21	*price*						money
							spend
							value
22	*N/A*						enough
							long
							trip
23	*N/A*						between
							difference
							make
							sure
24	*N/A*						package
							show
							travel
25	*price*						mid
							price
							range
26	*general terms*						curve
							custom
							function
							learning

Appendix B. Concept Clustering for DC data with Single-linkage Hierarchical Clustering (Terms in Concept Clusters)

	Product Feature Description	0.91	0.92	0.93	0.94	0.95	0.96
1	*spot metering*	metering	metering	metering	metering	metering	metering
		spot	spot	spot	spot	spot	spot
2	*white balance*	white	white	white	white	white	white
		balance	balance	balance	balance	balance	balance
				black	black	black	black
3	*red eye reduction*	red	red	red	red	red	red
		eye	eye	eye	eye	eye	eye
4	*lens*	wide	wide	wide	wide	wide	wide
		telephoto	telephoto	telephoto	telephoto	telephoto	telephoto
		angle	angle	angle	angle	angle	angle
5	*ease of use*	learning	learning	learning	learning	learning	learning
		curve	curve	curve	curve	curve	curve
6	*optical zoom*	zoom	zoom	zoom	zoom	zoom	zoom
		optical	optical	optical	optical	optical	optical
7	*battery life*	battery	battery	battery	battery	battery	battery
		life	life	life	life	life	life
		aa	aa	aa	aa	aa	aa
					charger	charger	charger
						rechargeable	rechargeable
							charge
							second
							frame
							per
							fps
8	*camera performance under low light*	light	light	light	light	light	light
		low	low	low	low	low	low
9	*video frame rate / continuous shooting*	second	second	second	second	second	
		frame	frame	frame	frame	frame	
		per	per	per	per	per	
						fps	
10	*shutter speed / aperture*	speed	speed	speed	speed	speed	speed
		shutter	shutter	shutter	shutter	shutter	shutter
			aperture	aperture	aperture	aperture	aperture
		lag	lag	lag	lag	lag	lag
							background
11	*Lcd screen*	lcd	lcd	lcd	lcd	lcd	lcd
		screen	screen	screen	screen	screen	screen
						view	view

(To be continued)

	Product Feature Description	0.91	0.92	0.93	0.94	0.95	0.96
12	*memory storage*	stick	stick	stick	stick	stick	stick
		memory	memory	memory	memory	memory	memory
		card	card	card	card	card	card
		mb	mb	mb	mb	mb	mb
		gb	gb	gb	gb	gb	gb
			reader	reader	reader	reader	reader
					cf	cf	cf
					computer	computer	computer
							sd
							download
13	*camera catergories*	point	point	point	point	point	point
		shoot	shoot	shoot	shoot	shoot	shoot
14	*general term*	under	under	under	under	under	under
		condition	condition	condition	condition	condition	condition
15	*general term*	range	range	range	range	range	range
		mid	mid	mid	mid	mid	mid
16	*picture quality (noise and iso)*	noise	noise	noise	noise	noise	noise
		iso	iso	iso	iso	iso	iso
						high	high
						higher	higher
						end	end
							anyone
							recommend
17	*picture quality (colour)*	color	color	color	color	color	color
		clarity	clarity	clarity	clarity	clarity	clarity
							detail
18	*kit lens*	lens	lens	lens	lens	lens	lens
		kit	kit	kit	kit	kit	kit
				mm	mm	mm	mm
19	*general term*		complaint	complaint	complaint	complaint	complaint
			biggest	biggest	biggest	biggest	biggest
20	*size*		fit	fit	fit	fit	fit
			purse	purse	purse	purse	purse
			pocket	pocket	pocket	pocket	pocket
							cam
							bag
21	*super macro mode*		macro	macro	macro	macro	macro
			super	super	super	super	super
22	*price*		$	$	$	$	$
			cost	cost	cost	cost	cost
23	*picture quality (indoor vs outdoor)*		outdoor	outdoor	outdoor	outdoor	outdoor
			indoor	indoor	indoor	indoor	indoor
24	*general term*			level	level	level	level
				adjust	adjust	adjust	adjust

(To be continued)

	Product Feature Description	0.91	0.92	0.93	0.94	0.95	0.96
25	*camera catergories*			digital	digital	digital	digital
				slr	slr	slr	slr
						eos	eos
						rebel	rebel
						xt	xt
							camera
26	*model name*			elph	elph	elph	elph
				series	series	series	series
27	*general term*			picture	picture	picture	picture
				taking	taking	taking	taking
28	*dust on sensor*			dust	dust	dust	dust
				clean	clean	clean	clean
					sensor	sensor	sensor
29	*model name*			sony	sony	sony	sony
				cybershot	cybershot	cybershot	cybershot
30	*model name*			rebel	rebel		
				xt	xt		
31	*general term*			anyone	anyone	anyone	
				recommend	recommend	recommend	
32	*auto mode*			auto	auto	auto	auto
				mode	mode	mode	mode
33	*price*				money	money	money
					spend	spend	spend
34	*texture*				feel	feel	feel
					solid	solid	solid
35	*general term*				up	up	up
					pick	pick	pick
36	*N\A*				photographer	photographer	photographer
					amateur	amateur	amateur
							professional
37	*general term*				carry	carry	carry
					around	around	around
38	*general term*					else	else
						everything	everything
39	*N\A*					went	went
						store	store
40	*N\A*					compare	compare
						pentax	pentax
41	*model name*					canon	canon
						powershot	powershot
42	*ease of use*					user	user
						friend	friend
43	*general term*					read	read
						review	review
44	*video*					movie	movie
						sound	sound
							video

(To be continued)

	Product Feature Description	0.91	0.92	0.93	0.94	0.95	0.96
45	*buttons*					important	important
						buttons	buttons
46	*outbox*						out
							box
47	*ease of use*						use
							easy
48	*print quality*						megapixel
							prints
49	*general term*						make
							sure
50	*picture quality*						clear
							crisp
51	*N\A*						best
							market
52	*general term*						month
							ago

Appendix C. Concept Clusters for Digital Camera data (Comparative Studies)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.91	Cut Thres = 0.95	EPS = 0.93	EPS = 0.96	Function 1	Function 2
1	***Spot metering***	metering spot	metering spot			metering spot	metering spot
	Recall	84.62%	84.62%			84.62%	84.62%
	Precision	68.75%	68.75%			68.75%	68.75%
	F-score	75.86%	75.86%			75.86%	75.86%
2	***White balance***	white balance	white balance black	white balance black	white balance black	white balance	white balance
	Recall	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
	Precision	51.28%	45.45%	45.45%	45.45%	51.28%	51.28%
	F-score	67.80%	62.50%	62.50%	62.50%	67.80%	67.80%
3	***Red eye reduction***	red eye	red eye			red eye	red eye
	Recall	100.00%	100.00%			100.00%	100.00%
	Precision	50.00%	50.00%			50.00%	50.00%
	F-score	66.67%	66.67%			66.67%	66.67%
4	***Lens***	wide telephoto angle	wide telephoto angle	wide telephoto angle	wide telephoto angle kit lens mm	wide telephoto angle kit lens mm efs	wide angle telephoto
	Recall	93.55%	93.55%	93.55%	97.97%	97.97%	93.55%
	Precision	78.38%	78.38%	78.38%	79.10%	78.14%	78.38%
	F-score	85.29%	85.29%	85.29%	87.53%	86.94%	85.29%
5	***General term***	learning curve	learning curve			learning curve	learning curve

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.91	Cut Thres = 0.95	EPS = 0.93	EPS = 0.96	Function 1	Function 2
6	***Optical zoom***	zoom optical	zoom optical			zoom optical	zoom optical
	Recall	100.00%	100.00%			100.00%	100.00%
	Precision	73.08%	73.08%			73.08%	73.08%
	F-score	84.44%	84.44%			84.44%	84.44%
7	***Battery life***	battery life aa	battery life aa charger rechargeable	battery life aa	battery life aa charger rechargeable charge fps frame per second	battery life aa charger rechargeable	battery life aa
	Recall	92.72%	96.03%	92.72%	98.01%	96.03%	92.72%
	Precision	90.32%	90.06%	90.32%	64.91%	90.06%	90.32%
	F-score	91.50%	92.95%	91.50%	78.10%	92.95%	91.50%
8	***Camera performance under low light***	light low	light low			light low	light low
	Recall	63.16%	63.16%			63.16%	63.16%
	Precision	23.08%	23.08%			23.08%	23.08%
	F-score	33.80%	33.80%			33.80%	33.80%
9	***Video frame rate / Continuous shooting***	second frame per	second frame per fps	second frame per		second frame per fps	second frame per
	Recall	46.15%	84.62%	46.15%		84.62%	46.15%
	Precision	21.43%	33.33%	21.43%		33.33%	21.43%
	F-score	29.27%	47.83%	29.27%		47.83%	29.27%

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.91	Cut Thres = 0.95	EPS = 0.93	EPS = 0.96	Function 1	Function 2
10	***Shutter speed / aperture***	speed	speed	speed	speed	speed	speed
		shutter	shutter	shutter	shutter	shutter	shutter
			aperture	aperture	aperture	aperture	aperture
		lag	lag	lag	lag	lag	lag
					background		
	Recall	81.82%	93.94%	93.94%	93.94%	93.94%	93.94%
	Precision	55.67%	59.05%	59.05%	55.36%	59.05%	59.05%
	F-score	66.26%	72.51%	72.51%	69.66%	72.51%	72.51%
11	***Lcd screen***	lcd	lcd		lcd	lcd	lcd
		screen	screen		screen	screen	screen
			view		view		
	Recall	96.46%	96.46%		96.46%	96.46%	96.46%
	Precision	87.90%	82.58%		82.58%	87.90%	87.90%
	F-score	91.98%	88.98%		88.98%	91.98%	91.98%
12	***Memory storage***	stick	stick	stick	stick	stick	stick
		memory	memory	memory	memory	memory	memory
		card	card	card	card	card	card
		mb	mb	mb	mb	mb	mb
		gb	gb	gb	gb	gb	gb
			reader	reader	reader	reader	reader
			cf	cf	cf	cf	cf
			computer		computer		
					download		
					sd		
	Recall	90.63%	90.63%	90.63%	92.71%	90.63%	90.63%
	Precision	67.97%	60.84%	67.44%	46.35%	67.44%	67.44%
	F-score	77.68%	72.80%	77.33%	61.81%	77.33%	77.33%
13	***Camera categories***	point	point			point	point
		shoot	shoot			shoot	shoot
14	***General term***	under	under			under	under
		condition	condition			condition	condition
15	***General term***	range	range			range	range
		mid	mid			mid	mid

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.91	Cut Thres = 0.95	EPS = 0.93	EPS = 0.96	Function 1	Function 2
16	***Picture quality (noise and iso)***	noise	noise		noise	noise	noise
		iso	iso		iso	iso	iso
			high		high		
			higher		higher		
			end		end		
					anyone		
					recommend		
	Recall	86.36%	86.36%		86.36%	86.36%	86.36%
	Precision	50.00%	14.84%		10.86%	50.00%	50.00%
	F-score	63.33%	25.33%		19.29%	63.33%	63.33%
17	***Picture quality (colour)***	color	color		color	color	color
		clarity	clarity		clarity	clarity	clarity
					detail		
	Recall	90.24%	90.24%		90.24%	90.24%	90.24%
	Precision	72.55%	72.55%		56.06%	72.55%	72.55%
	F-score	80.43%	80.43%		69.16%	80.43%	80.43%
18	***Kit lens***	lens	lens	lens			lens
		kit	kit	kit			kit
			mm	mm			mm
	Recall	93.98%	98.19%	98.19%			98.19%
	Precision	75.00%	71.18%	71.18%			71.18%
	F-score	83.42%	82.53%	82.53%			82.53%
19	***General term***		complaint			complaint	complaint
			biggest			biggest	biggest
20	***Size***		fit	fit	fit	fit	fit
			purse	purse	purse	purse	purse
			pocket	pocket	pocket		
					bag		
					cam		
	Recall		26.17%	26.17%	26.17%	85.29%	85.29%
	Precision		57.35%	57.35%	45.88%	61.70%	61.70%
	F-score		35.94%	35.94%	33.33%	71.60%	71.60%

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.91	Cut Thres = 0.95	EPS = 0.93	EPS = 0.96	Function 1	Function 2
21	***Super macro mode***		macro			macro	macro
			super			super	super
	Recall		51.69%			51.69%	51.69%
	Precision		80.70%			80.70%	80.70%
	F-score		63.01%			63.01%	63.01%
22	***Price***		$			$	$
			cost			cost	cost
	Recall		87.50%			87.50%	87.50%
	Precision		75.68%			75.68%	75.68%
	F-score		81.16%			81.16%	81.16%
23	***Picture quality (indoor vs outdoor)***		outdoor			outdoor	outdoor
			indoor			indoor	indoor
	Recall		87.50%			87.50%	87.50%
	Precision		75.68%			75.68%	75.68%
	F-score		81.16%			81.16%	81.16%
24	***General term***		level				
			adjust				
25	***Camera category***		digital				
			slr				
			eos				
			rebel				
			xt				
26	***Model name***		elph				
			series				
27	***General term***		picture				
			taking				

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.91	Cut Thres = 0.95	EPS = 0.93	EPS = 0.96	Function 1	Function 2
28	***Dust on sensor***		dust		dust		
			clean		clean		
			sensor		sensor		
	Recall		100.00%		100.00%		
	Precision		40.48%		40.48%		
	F-score		57.63%		57.63%		
29	***Model name***		sony				
			cybershot				
30	***General term***		anyone				
			recommend				
31	***Auto mode***		auto				
			mode				
	Recall		95.24%				
	Precision		14.81%				
	F-score		25.64%				
32	***Price***		money				
			spend				
	Recall		51.69%				
	Precision		80.70%				
	F-score		63.01%				
33	***Texture***		feel				
			solid				
	Recall		31.34%				
	Precision		40.38%				
	F-score		35.29%				
34	***General term***		up				
			pick				
35	***N\A***		photographer		photographer		
			amateur		amateur		
					professional		

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.91	Cut Thres = 0.95	EPS = 0.93	EPS = 0.96	Function 1	Function 2
36	***General term***		carry				
			around				
37	***General term***		else				
			everything				
38	***N\A***		went				
			store				
39	***N\A***		compare				
			pentax				
40	***Model name***		canon				
			powershot				
41	***Ease of use***		user				
			friend				
	Recall		10.81%				
	Precision		21.62%				
	F-score		14.41%				
42	***General term***		read				
			review				
43	***Video***		movie		movie		
			sound		sound		
					video		
	Recall		55.70%		93.67%		
	Precision		93.62%		92.50%		
	F-score		69.84%		93.08%		
44	***Buttons***		important				
			buttons				
	Recall		8.97%				
	Precision		30.43%				
	F-score		13.86%				

Appendix D. Concept Clusters for Personal Computer data (Comparative Studies)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.91	Cut Thres = 0.95	EPS = 0.93	EPS = 0.96	Function 3	Function 4
1	***Run parallel OS***	camp	camp		camp	camp	camp
		boot	boot		boot	boot	boot
					parallel	parallel	
					run		
	Recall	19.51%	19.51%		60.98%	21.95%	19.51%
	Precision	53.33%	53.33%		32.05%	56.25%	53.33%
	F-score	28.57%	28.57%		42.02%	31.58%	28.57%
2	***Hard Drive***	hard	hard		hard	hard	hard
		drive	drive		drive	drive	drive
					external	external	external
	Recall	91.53%	91.53%		93.22%	93.22%	93.22%
	Precision	60.00%	60.00%		54.46%	54.46%	54.46%
	F-score	72.48%	72.48%		68.75%	68.75%	68.75%
3	***CPU***	duo	duo	duo	duo	duo	duo
		core	core	core	core	core	core
		dual	dual	dual	dual	dual	dual
		processor	processor	processor	processor	processor	processor
		intel	intel	intel	intel	intel	intel
					faster		
						ghz	
	Recall	47.33%	47.33%	47.33%	50.00%	60.00%	47.33%
	Precision	82.56%	82.56%	82.56%	72.12%	84.91%	82.56%
	F-score	60.17%	60.17%	60.17%	59.06%	70.31%	60.17%
4	***Screen***	panel	panel			panel	panel
		flat	flat			flat	flat
	Recall	100.00%	100.00%			100.00%	100.00%
	Precision	25.00%	25.00%			25.00%	25.00%
	F-score	40.00%	40.00%			40.00%	40.00%

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.91	Cut Thres = 0.95	EPS = 0.93	EPS = 0.96	Function 3	Function 4
5	***CD-R / DVD-R***	movie	movie		movie	movie	movie
		watch	watch		watch	watch	watch
			dvd		dvd	dvd	
			cd		cd	cd	
			burn		burn	burn	
	Recall		85.00%		85.00%	85.00%	
	Precision		20.48%		20.48%	20.48%	
	F-score		33.01%		33.01%	33.01%	
6	***Mouse & keyboard***	mouse	mouse	mouse	mouse	mouse	mouse
		keyboard	keyboard	keyboard	keyboard	keyboard	keyboard
		wireless	wireless	wireless	wireless	wireless	wireless
		mighty	mighty	mighty	mighty	mighty	mighty
	Recall	92.24%	92.24%	92.24%	92.24%	92.24%	92.24%
	Precision	86.99%	86.99%	86.99%	86.99%	86.99%	86.99%
	F-score	89.54%	89.54%	89.54%	89.54%	89.54%	89.54%
7	***Power Supply***	power	power			power	power
		supply	supply			supply	supply
	Recall	86.11%	86.11%			86.11%	86.11%
	Precision	83.78%	83.78%			83.78%	83.78%
	F-score	84.93%	84.93%			84.93%	84.93%
8	***Memory (Ram)***	gb	gb	gb	gb	gb	gb
		ram	ram	ram	ram	ram	ram
		mb	mb	mb	mb	mb	mb
		upgraded	upgraded	upgraded	upgraded	upgraded	upgraded
			memory		memory	memory	
			ghz		ghz		ghz
					gig	gig	
	Recall	88.89%	99.07%	88.89%	99.07%	99.07%	88.89%
	Precision	83.48%	72.79%	83.48%	69.93%	78.68%	72.73%
	F-score	86.10%	83.92%	86.10%	81.99%	87.70%	80.00%

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.91	Cut Thres = 0.95	EPS = 0.93	EPS = 0.96	Function 3	Function 4
9	***Card Slot***	card slot pci graphic	card slot pci graphic	card slot pci graphic	card slot pci graphic	card slot pci graphic	card slot pci graphic
	Recall	81.40%	81.40%	81.40%	81.40%	81.40%	81.40%
	Precision	79.55%	79.55%	79.55%	79.55%	79.55%	79.55%
	F-score	80.46%	80.46%	80.46%	80.46%	80.46%	80.46%
10	***Multimedia Software***	video edit 	video edit itune imovie photo iphoto 	 	video edit itune imovie photo iphoto music	video edit itune imovie photo iphoto 	video edit imovie
	Recall	39.44%	74.65%		74.65%	74.65%	50.70%
	Precision	59.57%	55.79%		55.79%	55.79%	65.45%
	F-score	47.46%	63.86%		63.86%	63.86%	57.14%
11	***Multimedia Software***	itune iphoto photo imovie		itune iphoto photo imovie			itune iphoto photo
	Recall	74.65%		74.65%			74.65%
	Precision	64.63%		64.63%			64.63%
	F-score	69.28%		69.28%			69.28%
12	***General terms***	web site surf	web site surf	web site surf	web site surf	web surf site	web surf
13	***Office Software***	word process	word process			word process	word process
	Recall	49.02%	49.02%			49.02%	49.02%
	Precision	78.13%	78.13%			78.13%	78.13%
	F-score	60.24%	60.24%			60.24%	60.24%

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.91	Cut Thres = 0.95	EPS = 0.93	EPS = 0.96	Function 3	Function 4
14	***Connect wires***	port	port	port	port	port	port
		usb	usb	usb	usb	usb	usb
		firewire	firewire	firewire	firewire	firewire	firewire
						cable	
	Recall	84.21%	84.21%	84.21%	84.21%	90.24%	84.21%
	Precision	64.00%	64.00%	64.00%	64.00%	66.07%	64.00%
	F-score	72.73%	72.73%	72.73%	72.73%	76.29%	72.73%
15	***CD-R / DVD-R***	dvd		dvd			dvd
		cd		cd			burn
		burn		burn			cd
	Recall	75.00%		75.00%			75.00%
	Precision	32.61%		32.61%			32.61%
	F-score	45.45%		45.45%			45.45%
16	***General terms***	email	email			email	email
		mail	mail			mail	mail
17	***General terms***	customer	customer			customer	customer
		service	service			service	service
						called	
18	***Office Software***	microsoft	microsoft			microsoft	microsoft
		office	office			office	office
	Recall	56.86%	56.86%			56.86%	56.86%
	Precision	50.88%	50.88%			50.88%	50.88%
	F-score	53.70%	53.70%			53.70%	53.70%
19	***Speaker and Camera***	built	built	built	built	built	built
		speaker	speaker	speaker	speaker	speaker	speaker
		camera	camera	camera	camera	camera	camera
					digital		
	Recall	48.65%	48.65%	48.65%	48.65%	48.65%	48.65%
	Precision	41.86%	41.86%	41.86%	35.29%	41.86%	41.86%
	F-score	45.00%	45.00%	45.00%	40.91%	45.00%	45.00%

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.91	Cut Thres = 0.95	EPS = 0.93	EPS = 0.96	Function 3	Function 4
20	*N/A*	inside case	inside case			inside case	inside case
21	*Screen*	screen bright glossy	screen bright glossy	screen bright glossy	screen bright glossy clear size	screen glossy bright	screen glossy bright
	Recall	64.53%	64.53%	64.53%	66.28%	64.53%	64.53%
	Precision	92.50%	92.50%	92.50%	87.69%	92.50%	92.50%
	F-score	76.03%	76.03%	76.03%	75.50%	76.03%	76.03%
22	*Operating Systems*		window xp			window xp	window xp
	Recall		11.79%			11.79%	11.79%
	Precision		80.49%			80.49%	80.49%
	F-score		20.56%			20.56%	20.56%
23	*General terms*		speed slower			speed slower	speed slower
24	*General terms*		high recommend			high recommend	high recommend
25	*Printer*		hp printer		hp printer print	hp printer	hp printer
	Recall		84.21%		84.21%	84.21%	84.21%
	Precision		64.00%		57.14%	64.00%	64.00%
	F-score		72.73%		68.09%	72.73%	72.73%
26	*N/A*				bought year ago month		

Appendix E. Concept Clusters for Mobile data (Comparative Studies)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.93	Cut Thres = 0.94	EPS = 0.92	EPS = 0.94	Function 5	Function 6
1	***Wi fi***	wi fi	wi fi			wi fi	wi fi
	Recall	69.23%	69.23%			69.23%	69.23%
	Precision	87.10%	87.10%			87.10%	87.10%
	F-score	77.14%	77.14%			77.14%	77.14%
2	***Address book***	address book	address book			address book	address book
	Recall	83.33%	83.33%			83.33%	83.33%
	Precision	80.00%	80.00%			80.00%	80.00%
	F-score	81.63%	81.63%			81.63%	81.63%
3	***Ring tone***	ring tone	ring tone			ring tone	ring tone
	Recall	97.14%	97.14%			97.14%	97.14%
	Precision	68.00%	68.00%			68.00%	68.00%
	F-score	80.00%	80.00%			80.00%	80.00%
4	***Usb cable***	usb cable	usb cable			usb cable	usb cable
	Recall	100.00%	100.00%			100.00%	100.00%
	Precision	97.37%	97.37%			97.37%	97.37%
	F-score	98.67%	98.67%			98.67%	98.67%
5	***Battery life***	battery life	battery life			battery life	battery life
	Recall	84.21%	84.21%			84.21%	84.21%
	Precision	81.63%	81.63%			81.63%	81.63%
	F-score	82.90%	82.90%			82.90%	82.90%

(To be continued)

	Product Feature Description		Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
			Cut Thres = 0.93	Cut Thres = 0.94	EPS = 0.92	EPS = 0.94	Function 5	Function 6
6	***Software (pc suite)***		pc suite software	pc suite software	pc suite software	pc suite software connect	pc suite software	pc suite software
		Recall	52.70%	52.70%	52.70%	52.70%	52.70%	52.70%
		Precision	81.25%	81.25%	81.25%	69.64%	81.25%	81.25%
		F-score	63.93%	63.93%	63.93%	60.00%	63.93%	63.93%
7	***Noisy background***		noise background	noise background			noise background	noise background
		Recall	80.00%	80.00%			80.00%	80.00%
		Precision	70.59%	70.59%			70.59%	70.59%
		F-score	75.00%	75.00%			75.00%	75.00%
8	***Memory***		micro sd card gb microsd memory stick	micro sd card gb microsd memory stick	micro sd card gb microsd memory	micro sd card gb microsd memory stick	micro sd card gb microsd memory stick	micro sd card gb microsd memory stick
		Recall	98.02%	98.02%	98.02%	98.02%	98.02%	98.02%
		Precision	79.20%	79.20%	81.82%	79.20%	79.20%	79.20%
		F-score	87.61%	87.61%	89.19%	87.61%	87.61%	87.61%
9	***Radio***		radio fm station listen	radio fm station listen change	radio fm station	radio fm station listen change	radio fm	radio fm
		Recall	93.02%	93.02%	93.02%	93.02%	93.02%	93.02%
		Precision	57.97%	48.19%	70.18%	48.19%	76.92%	76.92%
		F-score	71.43%	63.49%	80.00%	63.49%	84.21%	84.21%

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.93	Cut Thres = 0.94	EPS = 0.92	EPS = 0.94	Function 5	Function 6
10	***Processor***	processor mhz	processor mhz			processor mhz	processor mhz
	Recall	89.47%	89.47%			89.47%	89.47%
	Precision	94.44%	94.44%			94.44%	94.44%
	F-score	91.89%	91.89%			91.89%	91.89%
11	***User friendly***	user friendly	user friendly			user friendly	user friendly
	Recall	100.00%	100.00%			100.00%	100.00%
	Precision	36.36%	36.36%			36.36%	36.36%
	F-score	53.33%	53.33%			53.33%	53.33%
12	***Keyboard***	keyboard qwerty full	keyboard qwerty full	keyboard qwerty full	keyboard qwerty full	keyboard qwerty full	keyboard qwerty full
	Recall	70.51%	70.51%	70.51%	70.51%	70.51%	70.51%
	Precision	71.43%	71.43%	71.43%	71.43%	71.43%	71.43%
	F-score	70.97%	70.97%	70.97%	70.97%	70.97%	70.97%
13	***MP3 player***	mp player music media	mp player music media listening	mp player music media	mp player music media listening	mp player	mp player
	Recall	95.73%	95.73%	95.73%	95.73%	66.67%	66.67%
	Precision	52.09%	54.37%	52.09%	54.37%	65.55%	65.55%
	F-score	67.47%	69.35%	67.47%	69.35%	66.10%	66.10%
14	***Sound quality***	sound quality	sound quality			sound quality	sound quality
	Recall	86.96%	86.96%			86.96%	86.96%
	Precision	36.36%	36.36%			36.36%	36.36%
	F-score	51.28%	51.28%			51.28%	51.28%
15	***General terms***	file transfer	file transfer			file transfer	file transfer

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.93	Cut Thres = 0.94	EPS = 0.92	EPS = 0.94	Function 5	Function 6
16	***General terms***	read reviews	read reviews			read reviews	read reviews
17	***General terms***	year contract	year contract			year contract	year contract
18	***General terms***	press button	press button			press button	press button
19	***Headphone adapter***	headphone adapter	headphone adapter			headphone adapter	headphone adapter
	Recall	91.43%	91.43%			91.43%	91.43%
	Precision	82.05%	82.05%			82.05%	82.05%
	F-score	86.49%	86.49%			86.49%	86.49%
19	***Earphone standard***	standard jack	standard jack			standard jack	standard jack
	Recall	89.47%	89.47%			89.47%	89.47%
	Precision	48.57%	48.57%			48.57%	48.57%
	F-score	62.96%	62.96%			62.96%	62.96%
20	***General terms***	customer service	customer service called		customer service called	customer service	customer service
21	***Application***	outlook calendar	outlook calendar contacts		outlook calendar contacts	outlook calendar	outlook calendar
	Recall	65.22%	86.96%		86.96%	65.22%	65.22%
	Precision	93.75%	90.91%		90.91%	93.75%	93.75%
	F-score	76.92%	88.89%		88.89%	76.92%	76.92%
22	***Size***	small size	small size			small size	small size
	Recall	88.89%	88.89%			88.89%	88.89%
	Precision	29.63%	29.63%			29.63%	29.63%
	F-score	44.44%	44.44%			44.44%	44.44%

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.93	Cut Thres = 0.94	EPS = 0.92	EPS = 0.94	Function 5	Function 6
23	***Software (windows mobile)***	windows mobile	windows mobile			windows mobile	windows mobile
	Recall	100.00%	100.00%			100.00%	100.00%
	Precision	22.78%	22.78%			22.78%	22.78%
	F-score	37.11%	37.11%			37.11%	37.11%
24	***General terms***	text message	text message			text message	text message
25	***Stereo & Headset***	headset stereo	headset stereo bluetooth		headset stereo bluetooth	headset stereo	headset stereo
	Recall	46.08%	46.08%		46.08%	46.08%	46.08%
	Precision	95.92%	95.92%		95.92%	95.92%	95.92%
	F-score	62.25%	62.25%		62.25%	62.25%	62.25%
26	***Display***	display bright	display bright			display bright	display bright
	Recall	29.23%	29.23%			29.23%	29.23%
	Precision	77.55%	77.55%			77.55%	77.55%
	F-score	42.46%	42.46%			42.46%	42.46%
27	***General terms***	days ago	days ago			days ago	days ago
28	***Digital camera***	digital camera	digital camera			digital camera	digital camera
	Recall	85.94%	85.94%			85.94%	85.94%
	Precision	93.22%	93.22%			93.22%	93.22%
	F-score	89.43%	89.43%			89.43%	89.43%
29	***General terms***	plan data	plan data			plan data	plan data

(To be continued)

	Product Feature Description		Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
			Cut Thres = 0.93	Cut Thres = 0.94	EPS = 0.92	EPS = 0.94	Function 5	Function 6
30	*Video recording*		video recording	video recording			video recording	video recording
		Recall	85.71%	85.71%			85.71%	85.71%
		Precision	67.92%	67.92%			67.92%	67.92%
		F-score	75.79%	75.79%			75.79%	75.79%
31	*General terms*			talking minute				
32	*General terms*			large fingers				

Appendix F. Concept Clusters for MP3 data (Comparative Studies)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.92	Cut Thres = 0.95	EPS = 0.95	EPS = 0.96	Function 7	Function 8
1	***Software (Sonic Stage)***	sonic	sonic	sonic	sonic	sonic	sonic
		stage	stage	stage	stage	stage	stage
			program	program	program		
	Recall	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
	Precision	95.74%	77.59%	77.59%	77.59%	95.74%	95.74%
	F-score	97.83%	87.38%	87.38%	87.38%	97.83%	97.83%
2	***Drag & drop***	drag	drag		drag	drag	drag
		drop	drop		drop	drop	drop
					media		
					window		
					explorer		
					sync		
	Recall	72.41%	72.41%	72.41%	72.41%	72.41%	72.41%
	Precision	63.64%	63.64%	63.64%	18.42%	63.64%	63.64%
	F-score	67.74%	67.74%	67.74%	29.37%	67.74%	67.74%
3	***General terms***	arm	arm			arm	arm
		band	band			band	band
4	***Radio***	fm	fm	fm	fm	fm	fm
		tuner	tuner	tuner	tuner	tuner	tuner
		radio	radio	radio	radio	radio	radio
			voice	voice	voice		
			recorder	recorder	recorder		
	Recall	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
	Precision	94.19%	64.29%	64.29%	64.29%	94.19%	94.19%
	F-score	97.01%	78.26%	78.26%	78.26%	97.01%	97.01%

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.92	Cut Thres = 0.95	EPS = 0.95	EPS = 0.96	Function 7	Function 8
5	***Voice Recorder***	voice recorder				voice recorder	voice recorder
	Recall	100.00 %				100.00%	100.00%
	Precision	54.55%				54.55%	54.55%
	F-score	70.59%				70.59%	70.59%
6	***Battery life***	battery life aaa	battery life aaa hours	battery life aaa hours	battery life aaa hours takes	battery life aaa hours	battery life aaa hours
	Recall	70.39%	78.29%	78.29%	80.26%	78.29%	78.29%
	Precision	96.40%	78.29%	78.29%	71.35%	78.29%	78.29%
	F-score	81.37%	78.29%	78.29%	75.54%	78.29%	78.29%
7	***Fast forward***	fast forward	fast forward		fast forward pause track beginning	fast forward	fast forward
	Recall	78.26%	78.26%		82.61%	78.26%	78.26%
	Precision	62.07%	62.07%		35.85%	62.07%	62.07%
	F-score	69.23%	69.23%		50.00%	69.23%	69.23%
8	***General terms***	customer service	customer service			customer service	customer service
9	***Hard drive***	hard drive	hard drive		hard drive memory flash	hard drive	hard drive
	Recall	82.22%	82.22%		86.67%	82.22%	82.22%
	Precision	49.33%	49.33%		35.78%	49.33%	49.33%
	F-score	61.67%	61.67%		50.65%	61.67%	61.67%

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.92	Cut Thres = 0.95	EPS = 0.95	EPS = 0.96	Function 7	Function 8
10	***Software (window media)***	media window explorer	media window explorer	media window explorer sync		media window explorer	media window explorer
	Recall	100.00%	100.00%	100.00%		100.00%	100.00%
	Precision	28.17%	28.17%	22.47%		28.17%	28.17%
	F-score	43.96%	43.96%	36.70%		43.96%	43.96%
11	***Sound quality***	sound quality	sound quality			sound quality	sound quality
	Recall	97.75%	97.75%			97.75%	97.75%
	Precision	67.97%	67.97%			67.97%	67.97%
	F-score	80.18%	80.18%			80.18%	80.18%
12	***Audio book***	audio book	audio book podcasts	audio book podcasts	audio book podcasts	audio book	audio book
	Recall	76.47%	82.35%	82.35%	82.35%	76.47%	76.47%
	Precision	23.64%	20.90%	20.90%	20.90%	23.64%	23.64%
	F-score	36.11%	33.33%	33.33%	33.33%	36.11%	36.11%
13	***DVR station***	dvr station	dvr station dock	dvr station dock	dvr station dock	dvr station	dvr station
	Recall	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
	Precision	52.78%	44.19%	44.19%	44.19%	52.78%	52.78%
	F-score	69.09%	61.29%	61.29%	61.29%	69.09%	69.09%
14	***Fast forward***	usb port cable connection supplied	usb port cable connection supplied	usb port cable connection supplied	usb port cable connection supplied	usb port	usb port
	Recall	96.30%	96.30%	96.30%	96.30%	77.78%	77.78%
	Precision	59.77%	59.77%	59.77%	59.77%	65.63%	65.63%
	F-score	73.76%	73.76%	73.76%	73.76%	71.19%	71.19%

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.92	Cut Thres = 0.95	EPS = 0.95	EPS = 0.96	Function 7	Function 8
15	***General terms***	movies tv show	movies tv show watch	movies tv show watch	movies tv show watch	movies tv show watch	movies tv show watch
16	***Firmware update***	firmware update	firmware update			firmware update	firmware update
	Recall	93.75%	93.75%			93.75%	93.75%
	Precision	81.08%	81.08%			81.08%	81.08%
	F-score	86.96%	86.96%			86.96%	86.96%
17	***General terms***	mp3 player	mp3 player			mp3 player	mp3 player
18	***General terms***	tech support	tech support			tech support	tech support
19	***General terms***	entire collection	entire collection			entire collection	entire collection
20	***General terms***	capacity storage	capacity storage			capacity storage	capacity storage
21	***Brand name***	zune marketplace	zune marketplace	zune marketplace	zune marketplace itunes	zune marketplace	zune marketplace
22	***Memory***	memory flash	memory flash			memory flash	memory flash
	Recall	91.18%	91.18%			91.18%	91.18%
	Precision	70.45%	70.45%			70.45%	70.45%
	F-score	79.49%	79.49%			79.49%	79.49%
23	***General terms***	package instructions	package instructions				

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.92	Cut Thres = 0.95	EPS = 0.95	EPS = 0.96	Function 7	Function 8
24	***Brand name***	zen	zen	zen	zen		
		vision	vision	vision	vision		
			creative	creative	creative		
25	***General terms***		month	month	month		
			ago	ago	ago		
			years	years	years		
26	***Button***		button	button	button		
			reset	reset	reset		
			push	push	push		
	Recall		93.67%	93.67%	93.67%		
	Precision		69.81%	69.81%	69.81%		
	F-score		80.00%	80.00%	80.00%		
27	***Format***		format	format	format		
			atrac	atrac	atrac		
			wma	wma	wma		
	Recall		97.73%	97.73%	97.73%		
	Precision		69.35%	69.35%	69.35%		
	F-score		81.13%	81.13%	81.13%		
28	***General terms***		give				
			star				
29	***Menu***		menu		menu		
			navigate		navigate		
					easy		
	Recall		90.57%		96.23%		
	Precision		88.89%		42.50%		
	F-score		89.72%		58.96%		
30	***Shuffle mode***		shuffle				
			mode				
	Recall		87.50%				
	Precision		15.91%				
	F-score		26.92%				

(To be continued)

	Product Feature Description	Hierarchical Clustering		DB Scan		Scalable Distance Clustering Algorithm	
		Cut Thres = 0.92	Cut Thres = 0.95	EPS = 0.95	EPS = 0.96	Function 7	Function 8
31	***General terms***		view				
			photos				
32	***General terms***		high				
			recommend				
33	***Pause***		track	track			
			beginning	beginning			
			pause	pause			
	Recall		32.00%	32.00%			
	Precision		26.67%	26.67%			
	F-score		29.09%	29.09%			
34	***General terms***		file				
			copy				
35	***General terms***		oled				
			bright				
36	***General terms***		web				
			browser				
37	***Scroll wheel***				scroll		
					wheel		
					click		
	Recall				85.71%		
	Precision				26.09%		
	F-score				40.00%		