COMBINING MACHINE-LEARNING AND SEMANTIC-ORIENTATION APPROACHES FOR SENTIMENT CLASSIFICATION

TITIMA KASEMSRITANAWAT

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR
THE DEGREE OF MASTER OF SCIENCE
(TECHNOLOGY OF INFORMATION SYSTEM MANAGEMENT)
FACULTY OF GRADUATE STUDIES
MAHIDOL UNIVERSITY
2013

Thesis
entitled

COMBINING MACHINE-LEARNING AND SEMANTIC-ORIENTATION APPROACHES FOR SENTIMENT CLASSIFICATION

Titima Kasemsritanawat
.....
Miss Titima Kasemsritanawat
Candidate

Tanasanee Phienthrakul
.....
Lect. Tanasanee Phienthrakul,
Ph.D. (Computer Engineering)
Major advisor

Mingmanas Sivaraksa
.....
Lect. Mingmanas Sivaraksa,
Ph.D. (Information Engineering)
Co-advisor

Waranyu Wongseree
.....
Lect. Waranyu Wongseree,
Ph.D. (Electrical Engineering)
Co-advisor

B. Mahaisavariya
.....
Prof. Banchong Mahaisavariya,
M.D., Dip Thai Board of Orthopedics
Dean
Faculty of Graduate Studies
Mahidol University

.....
Lect. Supaporn Kiattisin,
Ph.D. (Electrical and Computer Engineering)
Program Director
Master of Science Program in Technology of Information System Management
Faculty of Engineering
Mahidol University

Thesis
entitled

COMBINING MACHINE-LEARNING AND SEMANTIC-ORIENTATION APPROACHES FOR SENTIMENT CLASSIFICATION

was submitted to the Faculty of Graduate Studies, Mahidol University
for the degree of Master of Science
(Technology of Information System Management)
on
May 22, 2013

Titima kasemsritanawat
.....
Miss Titima Kasemsritanawat
Candidate

.....
Asst. Prof. Bunlur Emaruchi,
Ph.D. (Environmental Systems Engineering)
Chair

.....
Lect. Supoj Hengpraprohm,
Ph.D. (Computer Engineering)
Member

Tanasanee Phienthrakul
.....
Lect. Tanasanee Phienthrakul,
Ph.D. (Computer Engineering)
Member

.....
Lect. Mingmanas Sivaraksa,
Ph.D. (Information Engineering)
Member

Waranyu Wongseree
.....
Lect. Waranyu Wongseree,
Ph.D. (Electrical Engineering)
Member

.....
Prof. Banchong Mahaisavariya,
M.D., Dip. Thai Board of Orthopedics
Dean
Faculty of Graduate Studies
Mahidol University

.....
Lect. Worawit Israngkul,
M.S. (Technical Management)
Dean
Faculty of Engineering
Mahidol University

ACKNOWLEDGEMENTS

The success of this thesis became a reality with kindness and encouragement of my major advisor. I would like to express my sincere gratitude and the deepest appreciation to my major advisor, Dr. Tanasanee Phienthrakul for her supervision, valuable advice, revision of this thesis, manuscript resulting, numerous suggestion, and knowledge to do this thesis, both practice and theories which were so much beneficial for successful accomplishment of this study.

I also would like to express my deep appreciation to my co-advisor, Dr. Waranyu Wongseree who gives valuable advice, knowledge, and constructive comments. I am grateful to my co-advisor, Dr. Mingmanas Sivaraksa for her helpful suggestion. And, I sincerely thank to Dr. Supoj Hengpraprohm, who was the external examiner of the thesis defense, for his kindness and providing suggestions for improvement. I wish to thank Asst. Prof. Bunlur Emaruchi, the chairman of thesis defense, for giving his perspective on this thesis and generous comments.

Besides, I sincerely thank to my friends, who encourage me and force me to carry out this thesis successfully.

Finally, my thanks for their constant love and support go to my parents who care me throughout my whole life. They encourage me during doing this thesis. When this thesis was succeeded, my parents had been very glad.

Titima Kasemsritanawat

COMBINING MACHINE-LEARNING AND SEMANTIC-ORIENTATION APPROACHES FOR SENTIMENT CLASSIFICATION

TITIMA KASEMSRITANAWAT 5236782 EGTI/M

M.SC. (TECHNOLOGY OF INFORMATION SYSTEM MANAGEMENT)

THESIS ADVISORY COMMITTEE: TANASANEE PHIENTHRAKUL, Ph.D., MINGMANAS SIVARAKSA, Ph.D., WARANYU WONGSEREE, Ph.D.

ABSTRACT

This study proposed using new extracted features for sentiment classification. These new features were derived from SentiWordNet and they are called “sentiment features”. These features are described by the strength of positivity, negativity, and objectivity for each part-of-speech. Support vector machines (SVM), multinomial Naïve Bayes, and Decision Tree (J48) were used to classify the reviews on the extracted features. The set of proposed features and bag-of-words features were then evaluated on movie reviews. The experimental results showed that using a combination of bag-of-words and sentiment features gave the highest accuracy when SVM were applied. Furthermore, the accuracies of both SVM and multinomial Naïve Bayes on sentiment features were higher than the accuracy of the lexicon analysis in review classification.

KEY WORDS: SENTIMENT CLASSIFICATION / SENTIWORDNET / MOVIE REVIEWS / FEATURE EXTRACTION

126 pages

การรวมวิธีการเรียนรู้ด้วยเครื่องและความหมายทางภาษาเพื่อจำแนกความคิดเห็นตามความรู้สึก
COMBINING MACHINE-LEARNING AND SEMANTIC-ORIENTATION APPROACHES FOR SENTIMENT CLASSIFICATION

ฐิติมา เกษมศรีธนาวัฒน์ 5236782 EGTI/M

ว.ทม. (เทคโนโลยีการจัดการระบบสารสนเทศ)

คณะกรรมการที่ปรึกษาวิทยานิพนธ์: ธนัสนี เพียรตระกูล, Ph.D., มิ่งมานัส ศิวรักษ์, Ph.D., วรัญญู วงษ์เสรี, Ph.D.

บทคัดย่อ

งานวิจัยนี้นำเสนอการสร้างคุณลักษณะใหม่ขึ้นมาเพื่อใช้ในงานจำแนกความคิดเห็นตามความรู้สึก คุณลักษณะแบบใหม่นี้ได้มาจากเซนติเวิร์ดเน็ตและถูกเรียกว่า คุณลักษณะตามความรู้สึก คุณลักษณะนี้จะอธิบายถึงความแรงของความเป็นบวก ความเป็นลบ และความเป็นกลางของแต่ละประเภทของคำศัพท์ ซัพพอร์ตเวกเตอร์แมชชีน, เบย์อย่างง่าย (มัลติโนเมียล) และ ต้นไม้ตัดสินใจ (เจ 48) ถูกใช้เพื่อจำแนกความคิดเห็นบนคุณลักษณะตามความรู้สึก ทั้งคุณลักษณะใหม่นี้และคุณลักษณะโดยถุงคำศัพท์ได้ถูกประเมินความถูกต้องบนความคิดเห็นทางภาพยนต์ ผลการทดลองได้แสดงให้เห็นว่าเมื่อใช้คุณลักษณะที่ได้จากการรวมกันของคุณลักษณะโดยถุงคำศัพท์ กับคุณลักษณะตามความรู้สึกให้ความแม่นยำสูงสุดร่วมกับการใช้ ซัพพอร์ตเวกเตอร์แมชชีน นอกจากนี้ความแม่นยำในการจำแนกความคิดเห็นโดยใช้ ซัพพอร์ตเวกเตอร์แมชชีน และเบย์อย่างง่าย (มัลติโน-เมียล) บนคุณลักษณะตามความรู้สึกให้ความแม่นยำในการจำแนกความคิดเห็น มากกว่าการใช้การวิเคราะห์เชิงความหมายของคำศัพท์ด้วยพจนานุกรม

126 หน้า

CONTENTS

CONTENTS (cont.)

CONTENTS (cont.)

LIST OF TABLES

LIST OF FIGURES

CHAPTER I
INTRODUCTION

1.1 Background and Statement of Problems

Nowadays, there is a lot of information on the Internet. Users can search on the Internet for knowledge, information, or entertainment. Furthermore, there are many websites allowing the users to write reviews of products, service or movies. These reviews are useful for the readers who read these reviews to take decision to buy or choose one. However, if there are too many reviews and each review is long, the readers will take a lot of time to read all reviews. Also, the readers may have biases towards some reviews which may make misleading decision to buy or choose one.

In recent years a great deal of research has been done on categorizing documents. The categories could be based on subject, genre or the sentiment expressed in the document [1]. The classification of document into positive or negative orientation (sentiment classification) is used to summarize the reviews as "recommended" or "not recommended" so the readers can decide to buy or choose one at the beginning briefly.

The approach in sentiment classification is mainly divided into semantic approach and machine learning approach. Each approach also has advantages and disadvantages. The advantage of machine learning approach is gaining high accurate classification (up to 83% accurate) [1]. However, the disadvantage of machine learning approach is that features which are selected or extracted are necessary to be potential for classification. The extracted features should affect classification, while the advantage of semantic approach is reliable in linguistic point of view. However, usage of a lexicon only in semantic approach may gain low accuracy in classification.

Therefore, applying the advantage of semantic approach to machine learning technique may be an effective technique. In this study, new sentiment features which imply semantic orientation are extracted by using lexicon. These new

extracted features and bag-of-words features are used in classification. The bag-of-words features are brought from word unigrams in training set of dataset. In classifying documents, machine learning classifiers were applied in many researches and effective in text classification, i.e., Support Vector Machines (SVM) [2], Naïve Bayes (Multinomial) [3]. Decision tree (J48) [4] was also used as classifier to compare to those classifiers. Furthermore, there are many methods for improving performance of sentiment classification such as, dimension reduction, and constructing potential lexicon. In this study, Relief algorithm and PCA are used as dimensional reduction techniques for improving accuracy.

1.2 Objectives of Study

The objective of this study is to propose a technique of extracting features by using SentiWordNet, which is known as sentiment lexicon. The new extracted features are combined with bag-of-words features in order to improve accuracy of sentiment classification.

1.3 Scopes of Study

The scopes of this study are as follows:

1. The main approach for sentiment classification in this study is machine learning, i.e., SVM, Naïve Bayes (Multinomial), and decision tree (J48).

2. Movie reviews used as dataset in this study are brought from website: http://www.cs.cornell.edu/people/pabo/movie-review-data/ [5] and it is alternative version created by Nathan Treloar. This dataset composes of 692 negative and 694 positive processed reviews.

3. The Dataset in this study is divided into 2 groups, training set and test set. There are 1108 reviews in training set and 278 reviews in test set. 5-fold cross-validation is performed on training set for comparing accuracies among algorithms. Test set is used to evaluate generalization of algorithm on unseen data.

4. The SentiWordNet is used in this study as lexicon.

5. Accuracy, recall, precision, F-measure, and statistical test are used to evaluate algorithms.

1.4 Expected Results

The sentiment features using SentiWordNet which are used together with bag-of-words features will improve accuracy of sentiment classification.

CHAPTER II
LITERATURE REVIEW

2.1 SentimentClassification

Research in automatic text classification seeks to develop models for assigning category labels to new documents or document segments based on a training set of documents that have been pre-classified by domain experts. Most studies have focused on topic-based classification, i.e., classifying documents by subject or topic (e.g., education vs. entertainment). However, researchers are increasingly turning their attention to non-topic-based classification. Examples of non-topic-based classification include genre classification and sentiment classification.[6]

Genre classification is classifying documents according to the style of text in the document, such as, subjectivity/objectivity. Sentiment classification is classifying documents expressing opinions whether the document expresses a positive or negative opinion.

Moreover, in aspect of B.Liu [7], sentiment classification is similar to topic-based classification but also different from classic topic-based classification, which classifies documents into predefined topic class, e.g., politics, science, sports, etc. In topic-based classification, topic-related words are important. However, in sentiment classification, topic-related words are unimportant. Instead, sentiment words that indicate positive or negative opinions are important, e.g., great, excellent, amazing, horrible, bad, worst, etc. [7]

Sentiment classification can be performed at word level, sentence level, or document level. However, the classification is usually performed at document-level [7]. The approaches of sentiment classification at sentence level, and document level can be mainly divided into 2 categories, semantic-orientation approach and machine-learning approach.

2.1.1 Semantic-Orientation Approach

The main idea of semantic-orientation approach is to identify the semantic orientations of words. The semantic orientation of a word can be also called "polarity". The semantic orientation of a word indicates the direction that the word deviates from the norm for its semantic group. Words that encode a desirable state (e.g., beautiful, awesome) have a positive orientation, while words that represent an undesirable state have a negative orientation (e.g., disappointing). [8]

In sentiment classification, sentiment words and phrases are words and phrases that express positive or negative sentiments. The sentiment words are mostly adjectives and adverbs, but can be also verbs and nouns [7]. To classify a document, numbers of terms with negative and positive sentiment are used. If the document contains more positive than negative terms the document is assigned in positive class, and if the number of negative terms exceeds the number of positive terms the document is assigned in negative class.

Furthermore, sentiment orientations of words and phrases are widely measured as real number. In a document, total value of the sentiment orientations of the words or phrases is used to classify the document into positive or negative class. If the total value is more than zero, the document is assigned in positive class, and if the total value is less than zero, the document is assigned in negative class. Hence, sentiment words or phrases identifications are important. The evaluations of sentiment orientations of those sentiment words or phrases are also important.

There are basically two types of identification sentiment words (or phrases) with their sentiment orientations, corpus-based approaches and dictionary-based approaches[7]. Corpus-based approaches have the main purpose which is to find co-occurrence patterns of words to determine their sentiments. The famous research using this approach is the work of Turney[9]. The first step is to identify phrases that contain adjectives or adverbs. The second step is to estimate semantic orientation of each extracted phrase by using PMI-IR algorithm. PMI-IR uses Pointwise Mutual Information (PMI) and Information Retrieval (IR) to measure the similarity of pairs of words or phrases. A phrase is assigned a numerical rating by taking the mutual information between the given phrase and the word "excellent" and subtracting the mutual information between the given phrase and the word "poor". In

addition, PMI is the log value of measure of the degree of statistical dependence, or co-occurrence of words. The third step is to assign the given review to a class, recommended or not recommended, based on the average semantic orientation of the phrases extracted from the review.[9]

Dictionary-based approaches have the main point to use synonyms, antonyms, and hierarchies in WordNet to determine sentiments of words, such as the work of Hu et al.[8]. Their study aims to identify product features and user opinions on these features. Sentence which has one or more features is identified as opinion sentence. The semantic orientation of the opinion words in the opinion sentence is used to determine whether the opinion sentence is positive or negative. To obtain adjective words or opinion words in their study, a set of seed adjectives is manually come up as the seed list with known their orientations and the set of each seed adjective is grown by searching for their synonyms in WordNet.

Furthermore, the General Inquirer is also used, such as the work of Kenedy et al. [1], and Meena et al. [10]. The concept of contextual valence shifters was used in the study [1]. Sentiment or semantic orientations of words can also be called "valence". Their assumption was that the valence of individual word may be reversed, strengthened, or weakened by context, e.g., the presence of negations or intensifiers. Their study used the General Inquirer in order to identify polarity of words (positive or negative), as well as negations, overstatements, and understatement. The result is that the treatment of context valence shifters, especially using negation, gives better accuracy than the basic method which only counts positive and negative terms.

The work of Meena et al. [10] takes into account conjunction roles in sentences. Their study mentioned that conjunctions have a substantial impact on the overall sentiment of a sentence and concentrated on the effects of conjunctions and sentence constructions. Their study used POS-tagging and dependency trees to analyze the sentence constructs. The main point is to find the main clause in a sentence in order to decide the sentence level polarity. After the main clause is identified, the sentiment orientation of the sentence is performed by counting positive and negative terms in the main clause. The General Inquirer is used to identify negative and positive terms in their study. Furthermore, the WordNet is also used in the case that words are not found in the General Inquirer. The result of the

conjunction analysis improves the sentiment classification more than the basic method which only counts positive and negative terms.

Moreover, the SentiWordNet is available, and the usage of SentiWordNet was applied to another approach such as, machine learning[11]. The work of Dang et al. will be discussed in 2.1.2 machine-learning approach.

2.1.2 Machine-Learning Approach

This approach uses machine-learning algorithm such as, SVM, Naïve Bayes, and decision tree to train a sentiment classifier with training dataset. In text classification, the assumption of Salton et al. [12] is that a document can be characterized by the tokens or words. The most common representation formats of textual data for training on machine-learning algorithm are bag-of-words features. The document tokenization based on words is formed into a binary-value vector of words or a frequency vector of words. The binary-value vector of words indicates that each word in a given document is present or absent. On the other hand, the frequency vector of words indicates how often each word appears in a given document.

Since number of bag-of-words features is the number of distinct words/tokens in training dataset, if the number of documents is large, the vector space is also very high dimensions. To mitigate the negative effects of high dimensionality, stop word removal and stemming are often used in reducing the final number of words [13]. Stemming and stop word removal will be discussed in next section.

Apart from using one word as one feature which is called “unigram model”, usage of more than one word as one feature which is called “n-gram model” can also implement. The n-gram model is useful because of preservation some information left out of the bag of word unigrams. However, the unigram-based models are employed by many text classification systems and these simple features of the model perform fairly well compared with other complicated types of features [14]. Furthermore, other combinations of single word unigrams with word n-grams are also possible, and could be effective depending on the domain and type of mining problem. [15]

Early research in sentiment classification on movie reviews [16] examined the effectiveness of three machine-learning algorithms i.e., Naïve Bayes, Maximum

Entropy, and Support Vector Machines. Their study used bag-of-words features, i.e., unigrams and bigrams. The best performance is obtained by using SVM in combination with unigrams.

The work of Matsumoto et al. [17] was performed on the same dataset as the study of Pang et al. [16]. Their study used frequent word sub-sequences and dependency sub-trees from sentences in the dataset as features of SVM. These features are used due to preserve the word order and syntactic relations between words. In their study, these features are used in combination with bag-of-words features in various ways. The best accuracy is from the features based on dependency sub-trees in combination with bag of words.

Moreover, non-textual information from documents is useful to be used as features to text classification. In the study of Dang et al. [11], 3 types of features are used, content-free, content-specific, and sentiment features. Content-free features compose of lexical, syntactic, and structural feature which are non-textual information. Content-specific features consist of important keywords and phrases on certain topics. Their study used unigrams and bigrams as content-specific features. Sentiment features refer to sentiment words which are from semantic-orientation approach because sentiment words identification is performed. The better accuracy is attributed to the sentiment features which are added to feature set which content-free and content-specific features exist. Furthermore, the feature selection is used and the accuracy of classification is better than the one which the feature selection is not used.

In the study of Finn [18], the document is represented as vector of part-of-speech (POS) features due to the assumption that the POSstatistics would reflect the style of the language sufficiently for the learning algorithm to distinguish between different genre classes. To further purpose, the usage of POS tagging is also applied to this study because a word can have severalPOS tags which inform different senses. The usage of POS tagging gives the correct sense of the word in different context, i.e., verb, adjective, adverb, or noun sense. Consequently, words with correctly identified POS tags may affect the accuracy of classification. In sentiment classification, content words i.e., nouns, verbs, adjectives, and adverbs, especially the latter two types, are used as sentiment words because their main functions are to express meaning which can be positive, negative, or objective meaning [7],[19]. Therefore, this study uses 4

POS tags, i.e., verb, adjective, adverb, and noun respectively for extracting new features.

In this study, sentiment scores i.e., negative, positive, and objective scoresof each word are used. The POS of word is also used in order to receive the correct sense of that word as described above. Hence, each sentiment score of each POS may be potential feature for machine-learning algorithm to classify the document into positive or negative class. However, each word can be several meanings in the same POS so the average sentiment scores of all meanings of a word with the same POS are considered. Due to perform at document level, this study pays attention to the average sentiment scores which are average of negative, positive, and objective scores of each POS in a document.

2.2 Text Pre-Processing

For mining large document collections, it is necessary to pre-process the text documentsand store the information in a data structure, which is more appropriate for further processingthan a plain text file. Even though, meanwhile several methods exist that try toexploit also the syntactic structure and semantics of text, most text mining approachesare based on the idea that a text document can be represented by a set of words, i.e.a text document is described based on the set of words contained in it (*bag-of-words*representation).[20]

2.2.1Tokenization [21]

In order to obtain all words that are used in a given text, a **tokenization**process is required,i.e. a text document is split into a stream of words. In this step, spaces are recognized as word separators (in which case, multiple spaces are reduced to one space). Furthermore, the following four particular cases have to be considered: digits, hyphens, punctuation marks, and the case of the letters.

Numbers without a surrounding context are vague. Thus, in general numbers should be disregarded. However, there is a case which is considered that digits might appear mixed within a word. For instance, ‘510B.C.’ is a clearly

important word which is informative. Thus, a preliminary approach for treating digits in the text might be to remove all words containing sequences of digits.

Hyphens pose another difficult decision to the lexical analyzer. Breaking up hyphenated words might be useful due to inconsistency of usage. For instance, this allows treating ‘state-of-the-art’ and ‘state of the art’ identically. However, there are words which include hyphens as an integral part, for instance, gilt-edge, B-49, etc. The most suitable procedure seems to adopt a general rule and specify the exceptions on a case by case basis.

Normally, punctuation marks are removed entirely in the process of lexical analysis. While some punctuation marks are an integral part of the word, for instance ‘510B.C.’, removing them does not seem to have an impact because the risk of misinterpretation in this case is minimal.

The case of letters is usually not important because the lexical analyzer normally converts all the text to either lower or upper case. However, part of semantics might be lost due to case conversion. For instance, the words Bank and bank have different meanings.

2.2.2 Sentence Splitting

Sentence splitting is a deterministic consequence of tokenization: a sentence ends when a sentence-ending character, i.e. “.”, “!”, or “?”, is found which is not grouped with other characters into a token (such as for an abbreviation or number). Though, it may still include a few tokens that can follow a sentence ending character as part of the same sentence (such as quotes and brackets). [22]

2.2.3 Part-of-Speech Tagging

Part-of-speech (POS) tagging is one of syntactic analyses. Tagging text with parts of speech is extremely useful for more complicated NLP tasks such as parsing and machine translation [23].POS tagging is the process of tagging words in a text with each word's corresponding part of speech. POS tagging is based both on the meaning of the word and its positional relationship with adjacent words. A simple list of the parts of speech for English includes adjective, adverb, conjunction, noun,

preposition, pronoun, and verb. For example, given the sentence “The kid is smart.” , the POS tagger would output “The/DT kid/NN is/VB smart/JJ ./.”.

The actual set of tags used in POS taggers is complex. There are three commonly used training datasets or tag sets: the Brown tag set, the C5 tag set, and the Penn Treebank tag set. The Penn Treebank tag set composes of 45 tags. There are 4 classes of POS tagger: rule-based, probabilistic-based, transformation-based, and maximum entropy-based taggers.

2.2.4 Elimination of Stop Words

Words which are too frequent among the documents in the collection are not good discriminators. Such words are frequently referred to as stop words and are normally filtered out, such as articles, prepositions, and conjunctions [21].Elimination of stop words has an important benefit which is reducing the size of feature vectors. However, stop word lists may be variant due to the purpose of works which have different concerns of the same word.

2.2.5 Stemming

The concept of stemmingis mapping several morphological forms of words to a common feature. The common way of stemming is to remove suffixes from words. For example the words *connected*, *connecting*, *connection*, and *connections* would all map to the common root *connect*, and this latter string would be placed in the feature set rather than the former four [14]. Thus, stemming makes the features more statistically independent. While there are three or four well known stemming algorithms, the most popular one is that by Porter algorithm because of its simplicity and elegance. [21]

2.3 Machine Learning for Classification

The general problem of machine learning is to search a, usually very large, space of potential hypotheses to determine the one that will best fit the data and any prior knowledge [24]. The data may be labeled or unlabeled. If labels are given then the problem is one of supervised learning in that the true answer is known for a given set of data. If the labels are categorical then the problem is one of classification, e.g. predicting the species of a flower given petal and sepal measurements[25].If the labels are real-valued the problem is one of regression, e.g. predicting property values from crime, pollution, etc. statistics[26].If labels are not given then the problem is one of unsupervised learning and the aim is characterize the structure of the data, e.g. by identifying groups of examples in the data that are collectively similar to each other and distinct from the other data. [27]

2.3.1 Naïve Bayes

Simple Bayesian classifiers have been gaining popularity, and have been found to perform surprisingly well. The naïve Bayes classifier is the simplest of these probabilistic models, in that it assumes that all attributes of the examples are independent of each other given the context of the class. This is the so-called "naïve Bayes assumption". Because of the independent assumption, the parameters for each attribute can be learned separately, and this greatly simplifies learning, especially when the number of attributes is large. Document classification is a domain with a large number of attributes. The attributes of the examples to be classified are words, and the number of distinct words can be quite large. While some simple document classification tasks can be accurately performed with vocabulary sizes less than one hundred, many complex tasks on real-world data from the Web do best with vocabulary sizes in the thousands.[3],[24]

The Bayesian approach to classifying the new instance is to assign the most probable target value or output with maximum a posteriori hypothesis (MAP), v_{MAP}, given the attribute values $\langle a_1, a_2, \dots, a_n \rangle$ that describe the instance.

$$v_{MAP} = \underset{v_j \epsilon V}{argmax} P(v_j | a_1, a_2, \dots, a_n) \quad (2.1)$$

Using Bayes theorem to rewrite this expression as

$$v_{MAP} = \underset{v_j \epsilon V}{argmax} \frac{P(a_1, a_2, \ldots, a_n | v_j) P(v_j)}{P(a_1, a_2, \ldots, a_n)}$$

$$= \underset{v_j \epsilon V}{argmax} P(a_1, a_2, \ldots, a_n | v_j) P(v_j) \quad (2.2)$$

The two terms in Equation (2.2) is to be estimated based on training data. $P(v_j)$is simply estimated by counting the frequency with which each target value v_j occurs in the training data. However, estimating the different $P(a_1, a_2, \ldots, a_n | v_j)$ terms in this fashion is not feasible. Thus, the naïve Bayes classifier is based on the simplifying assumption that the attribute values are conditionally independent given the target value. In other words, the assumption is that given the target value of the instance, the probability of observing the conjunction $a_1, a_2, \ldots, a_n$ is just the product of the probabilities for the individual attributes: $P(a_1, a_2, \ldots, a_n | v_j) = \prod_{i=1}^{n} P(a_i | v_j)$. Substituting this into (2.2), the naïve Bayes classifier is used by

$$v_{NB} = \underset{v_j \epsilon V}{argmax} P(v_j) \prod_{i=1}^{n} P(a_i | v_j) \quad (2.3)$$

Where v_{NB}denotes the target value output by the naïve Bayes classifier. Whenever the naïve Bayes assumption of conditional independence is satisfied, this naïve Bayes classification is identical to the MAP classification.[24]

Moreover, there are two different generative models in naïve Bayes classifier, multi-variate Bernoulli and multinomial model. The experiment of McCallum [3] showed that the multi-variate Bernoulli model sometimes performs better than the multinomial model at small vocabulary sizes. However, the multinomial usually outperforms the multi-variate Bernoulli at large vocabulary sizes, and almost always beats the multi-variate Bernoulli when vocabulary size is chosen optimally for both.

2.3.2 Decision Tree

Decision rules and decision tree based approaches to learning from text are particularly appealing, since rules and trees provide explanatory insight to end-users and text application developers [4]. Also, in the study of Jin-Cheon Na et al. [6] referred that SVM worked better than decision tree induction on the text classification. However, a decision tree model is easy to interpret and can be converted to IF-THEN rules.[6]

The basic algorithm, ID3, learns decision trees by constructing the tree top-down. Each instance attribute is evaluated using a statistical test to determine how well the attribute alone classifies the training examples. A best attribute is selected and used as the test at the root node of the tree. A descendant of the root note is then created for each possible value of this attribute, and the training examples are sorted to find the appropriate descendant node. The entire process is then repeated using the training examples associated with each descendant node to select the best attribute to test at that point in the tree. Thus, the main point is to select the attribute that is most useful for classifying examples. A good quantitative measure which has a statistical property, called information gain, measures how well a given attribute separates the training examples according to their target classification. In casual mention, the descendant node with highest information gain will become an attribute at this node.

In order to define information gain precisely, a measure commonly used in information theory is defined first, called entropy, that characterizes the impurity of an arbitrary collection of examples. Given a collection S, containing positive and negative examples of some target concept, the entropy of S relative to this Boolean classification is

$$Entropy(S) \equiv -p_{\oplus} log_2 p_{\oplus} - p_{\ominus} log_2 p_{\ominus} \quad (2.4)$$

where $p_{\oplus}$ is the proportion of positive examples in S and $p_{\ominus}$ is the proportion of negative examples in S. In all calculations involving the entropy, $0log0$, its value will be 0.

To illustrate, suppose Sis a collection of 14 examples of some boolean concept, including 9 positive and 5 negative examples (expressed as [9+, 5-]). Then the entropy of S relative to this Boolean classification is

$$Entropy([9+,5-]) = -\left(\frac{9}{14}\right) log_2 \left(\frac{9}{14}\right) - \left(\frac{5}{14}\right) log_2 (\frac{5}{14}) \quad (2.5)$$
$$= 0.940$$

More generally, if the target attribute can take on c different values, then the entropy of S relative to this c-wise classification is defined as

$$Entropy(S) \equiv \sum_{i=1}^{c} -p_i \, log_2 p_i \quad (2.6)$$

Where p_i is the proportion of S belonging to class i.

After entropy is obtained, a measure of the effectiveness of an attribute in classifying the training data (called information gain) is defined. Information gain is the expected reduction in entropy caused by partitioning the examples according to their values corresponding to the candidate attribute. More precisely, the information gain, $Gain(S, A)$ of an attribute A, relative to a collection of examples S, is defined as

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \epsilon Value(A)} \frac{|S_v|}{|S|} \, Entropy(S_v) \quad (2.7)$$

Where Value(A) is the set of all possible values for attribute A, and S_v is the subset of S for which attribute A has value v. Note the first term in Equation (2.7) is just the entropy of the original collection S, and the second term is the expected value of the entropy after S is partitioned as many subset using attribute A. The expected entropy described by this second term is simply the sum of the entropies of each subset S_v , weighted by the fraction of the examples $\frac{|S_v|}{|S|}$ that belong to S_v. The value of $Gain(S, A)$ is the number of bits saved when encoding the target value of an arbitrary member of *S*, by knowing the value of attribute A.

In growing the tree, information gain is precisely the measure to select the best attribute for classifying the training examples at each step. The candidate attribute with highest information gain will be selected. After the decision attribute at the root node are created, the branches with new nodes are created below the root for each of its possible values. The training examples will be sorted to new descendant node. If every example for new node which is corresponding to one of particular values belongs to the same target value or class, this new node of the tree becomes a leaf node with the classification, the target value. In contrast, if the new node which is corresponding to one of particular values has nonzero entropy, the decision tree will be further elaborated below this node.

The process of selecting a new attribute (or node) and partitioning examples is now repeated for each nonterminal descendant node, this time using only the training examples associated with that attribute. Attributes that have been incorporated higher in the tree are excluded, thus any given attribute can appear at most once along any path through the tree. This process continues for each new descendant node until either of two conditions is met: (1) every attribute has already been included along this path through the tree, or (2) the training examples associated with the node which is corresponding to one of particular values from the antecedent node have the same target value or class.[24]

J48 is a version of an earlier algorithm developed by J. Ross Quinlan, the very popular C4.5.The J48 algorithm gives several options related to tree pruning.Pruning produces fewer, more easily interpreted results. More importantly, pruning can be used as a tool to correct for potential overfitting.The basic algorithm described above recursively classifies until each leaf is pure, meaning that the data has been categorized as close to perfectly as possible. This process ensures maximum accuracy on the training data, but this process may create excessive rules that only describe particular idiosyncrasies of that data. When tested on new data, the rules may be less effective.Pruning always reduces the accuracy of a model on training data. This is because pruning employs various means to relax the specificity of the decision tree, hopefully improving its performance on test data. The overall concept is to gradually generalize a decision tree until it gains a balance of flexibility and accuracy.J48 employs two pruning methods. The first is known as subtree

replacement. This means that nodes in a decision tree may be replaced with a leaf; basically reducing the number of tests along a certain path. This process starts from the leaves of the fully formed tree, and works backwards toward the root.The second type of pruning used in J48 is termed subtreeraising. In this case, a node may be moved upwards towards the root of the tree, replacing other nodes along the way. Subtreeraising often has a negligible effect on decision tree models. There is often no clear way to predict the utility of the option, though it may be advisable to try turning it off if the induction process is taking a long time. This is due to the fact that subtreeraising can be somewhat computationally complex.[28]

2.3.3 Support Vector Machine (SVM)

SVM is very universal learner. In its basic form, SVM learns linear threshold function. Nevertheless, by a simple "plug-in" of an appropriate kernel function, SVM can be used to learn polynomial classifiers, radial basic function (RBF) networks, and three-layer sigmoid neural nets.

SVM work well for text classification due to the properties of text which is according to these followings: [2]

1. High dimensional input space: When learning text classifiers, one has to deal with very many (more than 10,000) features. Since SVM uses overfitting protection, which does not necessarily depend on the number of features, SVM has the potential to handle these large feature spaces.

2. Few irrelevant features: One way to avoid these high dimensional input spaces is to assume that most of the features are irrelevant. Feature selection tries to determine these irrelevant features. Unfortunately, in text classification there are only very few irrelevant features. Thus, those features seem unlikely completely redundant.

3. Document vector are sparse: For each document, the corresponding document vector contains only few entries which are not zero. Kivinen et al. [2],[29] give both theoretical and empirical evidence for the mistake bound model that "additive" algorithms, which have a similar inductive bias like SVM, are well suited for problems with dense concepts and sparse instances.

4. Most text classification problems are linearly separable: The idea of SVM is to find such linear (or polynomial, RBF, etc.) separators.

Two key elements in the implementation of SVM are the techniques of mathematical programming and kernel functions. The parameters are found by solving a quadratic programming problem with linear equality and inequality constraints. Due to focus on SVMs for two-class classification, the classes being P, N for y_i=+1,-1 respectively. Support vectors are the data points that lie closest to the separating hyperplane. The geometrical interpretation of support vector classification (SVC) is that the algorithm searches for the maximal margin hyperplane which is also equidistant from the two classes. The maximal margin hyperplane is shown in Figure 2.2. The support vectors are shown in Figure 2.1 and 2.2. SVC is outlined first for the linearly separable case. Kernel functions are then introduced in order to construct non-linear decision surfaces.[27]

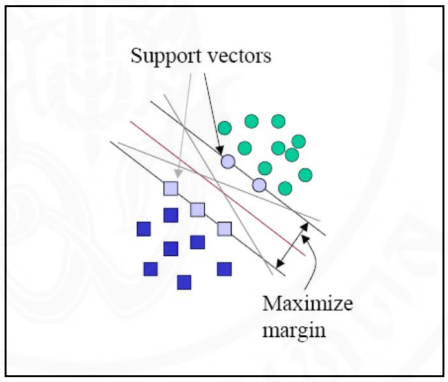


Figure 2.1 Show Support Vectors [30]

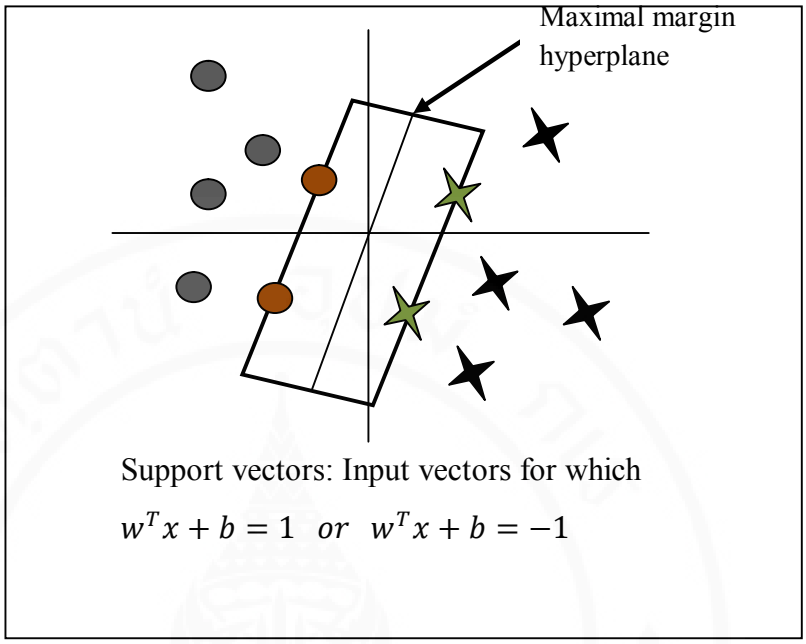


Figure 2.2 Show Support Vectors with Their Propertiesand Maximal Margin Hyperplane(Adjusted from [30])

If the training data are linearly separable then there exists a pair (w, b)such that

$$w^T x_i + b \geq 1 \quad , for\ all\ x_i \in P$$
$$w^T x_i + b \leq -1\ , for\ all\ x_i \in N \tag{2.8}$$

with the decision rule given by

$$f_{w,b}(x) = sign(w^T x + b). \tag{2.9}$$

$w$is termed the weight vector and b the bias. The inequality constraints (2.8) can be combined to give

$$y_i(w^T x_i + b) \geq 1, for\ all\ x_i \in P \cup N \tag{2.10}$$

In order to restrict the expressiveness of the hypothesis space, the SVM searches for the simplest solution that classifies the data correctly. The learning problem is hence reformulated as: minimize$\|w\|^2 = w^T w$ subject to the constraints of linear separability (2.10). This is equivalent to maximizing the distance, normal to the

hyperplane, between the convex hulls of the two classes; this distance is called the margin. The optimization is now a convex quadratic programming (QP) problem.

$$\min_{w,b} \Phi(w) = \frac{1}{2}\|w\|^2 \qquad (2.11)$$
$$\text{Subject to} \quad y_i(w^T x_i + b) \geq 1, i = 1, \ldots, l.$$

The Lagrangian for (2.11) is set. $\Lambda = (\lambda_i, \ldots \lambda_l)^T$ are the Lagrange multipliers, one for each data point. After the processing of QP solver is finished, the weight vector is obtained which is

$$w^* = \sum_{i=1}^{l} \lambda^* y_i x_i. \qquad (2.12)$$

The Lagrange multipliers are only non-zero when $y_i(w^T x_i + b) = 1$, vectors x_i in this case are called support vectors, seen in Figure 2.2, since these vectors lie closest to the separating hyperplane. The optimal weights are given by (2.12) and the bias is given by

$$b^* = y_i - w^{*T} x_i \qquad (2.13)$$

for any support vector x_i. The decision function is then given by

$$f(x) = sign\left(\sum_{i=1}^{l} y_i \lambda_i^* x^T x_i + b\right). \qquad (2.14)$$

The SVM can be used to learn non-linear decision functions by first mapping the data to some higher dimensional feature space and constructing a separating hyperplane in this space. Kernel-Induced Feature Spaces is applied to this idea.

Denoting the mapping to feature space by

$$X \to H$$
$$x \longmapsto \phi(x)$$

the decision function (2.9), and (2.14) become

$$f(x) = sign\left(\sum_{i=1}^{l} y_i \lambda_i^* \phi(x)^T \phi(x_i) + b^*\right). \quad (2.15)$$

The decision function (2.14) only in the form of inner products $x^T z$, and in the decision function (2.15) only in the form of inner products $\phi(x)^T \phi(z)$.Mapping the data to H is time consuming and storing it may be impossible, e.g. if H is infinite dimensional. Since the data only appear in inner products, computable function is required that gives the value of the inner product in H without explicitly performing the mapping.Hence, introduce a kernel function,

$$K(x, z) \equiv \phi(x)^T \phi(z) \quad (2.16)$$

The kernel function allows users to construct an optimal separating hyperplane in the space H without explicitly performing calculations in this space. Instead of calculating inner products, the value of K is computed. This requires that K is an easily computable function. The polynomial kernel is $K(x, z) = (x^T z + 1)^d$. The radial-basis function kernel is $K(x, z) = \exp\left(-\frac{\|x-z\|^2}{2\sigma^2}\right)$. The sigmoid kernel is $K(x, z) = \tanh(\beta_0 x^T z + \beta_1)$. The decision function (2.15) becomes

$$f(x) = sign(\sum_{i=1}^{l} y_i \lambda_i^* K(x, x_i) + b^*) \quad (2.17)$$

Where the bias is given by

$$b^* = y_i - x^{*T}\phi(x_i) = y_i - \sum_{j=1}^{l} y_j \lambda_i^* K(x_j, x_i) \quad (2.18)$$

for any support vector x_i.

2.4 Dimensional Reduction

In real-world concept learning problems, the representation of data often uses many features, only a few of which may be related to the target concept. In this situation, dimensional reduction is important both to speed up learning and to improve concept quality. There are many methods for dimensional reduction, such as Principal Component Analysis (PCA), and Relief Algorithm. PCA transforms original variables to a new set of variables which has smaller size than the original one. Furthermore, dimensional reduction method can select a small subset of features which is necessary and sufficient to describe the target concept, which is called "feature selection", such as Relief Algorithm.[31]

2.4.1 Principal Component Analysis (PCA)

The central idea of principal component analysis (PCA) is to reduce the dimension of a data set consisting of a large number of interrelatedvariables, while retaining as much as possible of the variation present inthe data set.This is achieved bytransforming to a new set of variables,the principal components (PCs), which are uncorrelated, and which areordered so that the first fewretain most of the variation present in allofthe original variables [32]. The main advantage of PCA is thatreducing the number of dimensions but withoutmuch loss of information.[33]

There are many elementary background mathematical skills thatwill be required to understand the process of PCA, such as, statistics which looks at distribution measurements,or, how the data is spread out. Standard deviation and variance are measure of the spread of data in data set. Moreover, Matrix Algebrais involved in PCA and the further process after receiving matrix algebra is looking for eigenvectors and eigenvalues, which are as important properties of matrices.

Due to operate onlyon 1 dimension for standard deviation and variance, the ones could be calculated for each dimension of the data set independently of the other dimensions. However, there is a similar measure which is useful to find out howmuch the dimensions vary from the mean with respect to each other.Covariance is such a measure.Covariance is measured between 2 dimensions.Since the covariancevalue can be calculated between any 2 dimensions in a data set, this techniqueis often used to find relationships between 2 dimensions in high-dimensional data sets where visualization is difficult.A useful way to get all the possible covariance values between all the differentdimensions is to calculate them all and put them in a matrix. If data set has n dimensions,then the matrix has n rows and n columns (so is square), and called "covariance matrix", and each entry in the matrix isthe result of calculating the covariance between two separate dimensions.

The point of view is to gain eigenvectors for covariance matrix of a data set. Another important thing is to make the eigenvectors to have their length are exactly 1 in order to keep eigenvectors standard. All the eigenvectors of the matrix areperpendicular,i.e., at right angles to each other, no matter how many dimensions there are. After gaining the eigenvectors, data can be expressed in terms of these perpendicular eigenvectors,instead of expressing them in terms of the x,y, z, etc axes. Furthermore, eigenvectors and eigenvalues always come in pairs.Eigenvalues are scalar which comes from multiplication of the covariance matrix and the same original vector followed by rearranging result into the same original vector multiples with any scalar which are referred as eigenvalues.

To perform a PCA on a set of data, there are these following steps and illustrated examples: [33]

1. Get some data: Suppose that there is 2-dimensional data setwhich is shown in Table 2.1 and data are plotted in 2 x-y axes as shown in Figure 2.3.

2. Subtract the mean: For PCA to work properly, the mean subtracts each of the data dimensions.The mean subtracted is the average across each dimension. Thus, all the x valueshave $\bar{x}$ (the mean of the x values of all the data points) subtracted, and all the y valueshave $\bar{y}$subtracted from them. This produces a data set whose mean is zero.The adjusted data are shown in Table 2.2.

Table 2.1 Example of Data Set [33]

x	y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9

Table 2.2 Adjusted Data [33]

x	y
.69	.49
-1.31	-1.21
.39	.99
.09	.29
1.29	1.09
.49	.79
.19	-.31
-.81	-.81
-.31	-.31
-.71	-1.01

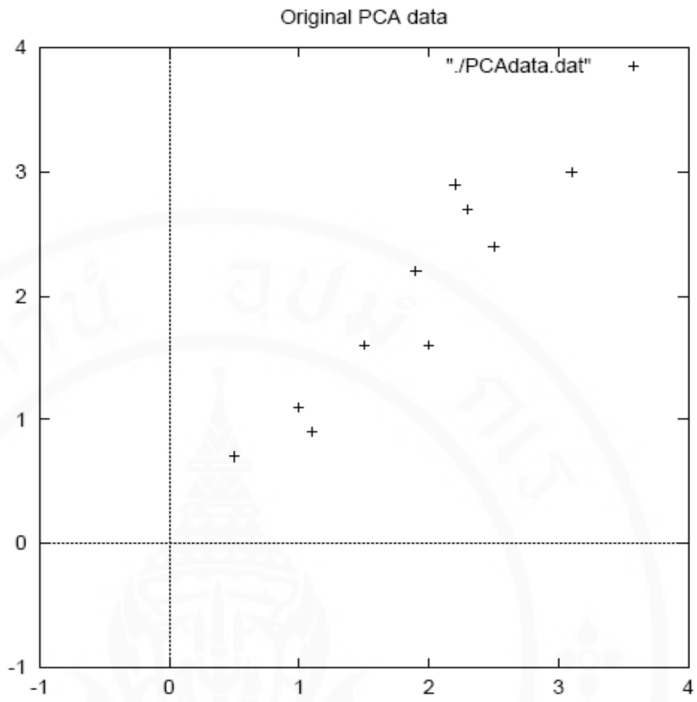


Figure 2.3 Original Data[33]

3. Calculate the covariance matrix: Since the data is 2-dimensional, the covariance matrix will be 2*2.

4. Calculate the eigenvectors and eigenvalues of the covariance matrix and the results are (2.19), and (2.20): Since the covariance matrix is square, the eigenvectors and eigenvalues can be calculated for this matrix. These eigenvectors as shown in Figure 2.4 both are unit eigenvectors. [33]

$$eigenvalues = \begin{pmatrix} .0490833989 \\ 1.28402771 \end{pmatrix} \quad (2.19)$$

$$eigenvectors = \begin{pmatrix} -.735178656 & -.677873399 \\ .677873399 & -.735178656 \end{pmatrix} \quad (2.20)$$

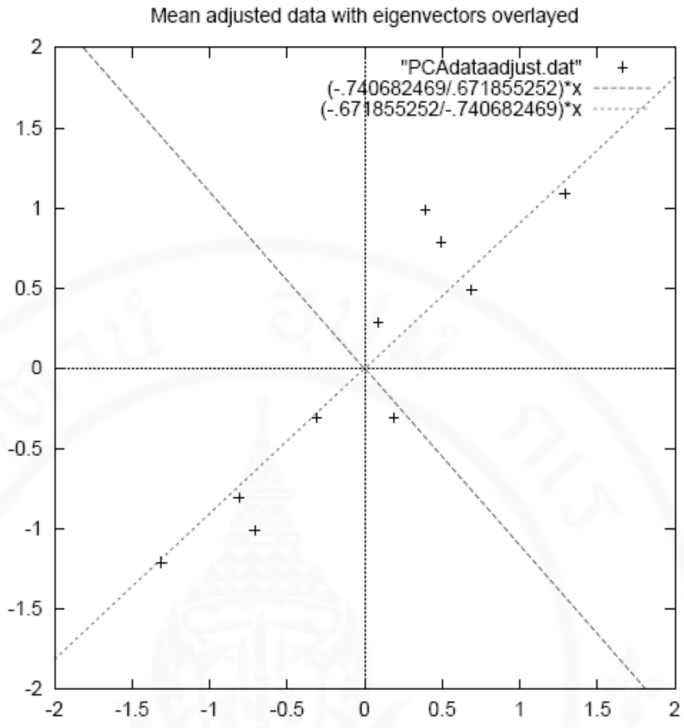


Figure 2.4 A Plot of the Normalized Data (Mean Subtracted) with the Eigenvectors of the Covariance Matrix Overlaid on Top. [33]

In Figure 2.4, on top of the data the eigenvectors both are plotted, which are perpendicular to each other.Theeigenvectorsprovide information about the variability in the data set. One of theeigenvectors goes through the middle of the points, like drawing a line of best fit. That eigenvector is showing how these two data sets are related along that line. The second eigenvector gives less important patterns in the data.

5. Choosing components and forming a feature vector: The eigenvector with the highest eigenvalue is the principle component of the data set. In the example, the eigenvector with the largest eigenvalue was the one that pointed down the middle of the data in Figure 2.4. This eigenvector is the most significant relationship between the data dimensions. In general, once eigenvectors are found from the covariance matrix, the next step is to order them by eigenvalue, highest to lowest. The component of lesser significance can be ignored but some information of data can lose. However, if the eigenvalues are small, the information of data doesn't lose much. If some components are left out, the final data set will have less dimensions than the original.

Before the process of PCA, there are n dimensions, but after the process, there are p dimensions or eigenvectors. Thus, feature of vector are as (2.21):

$$FeatureVector = (eig_1, eig_2, \dots, eig_p) \qquad (2.21)$$

In the example, there are 2 eigenvectors. If the small and less significant component is left out, there is only one component.

In the case of the transformation data using botheigenvectors,the altered data as shown in Table 2.3 is simply in terms of those eigenvectors instead of the usual axes. The data are plotted in term of those two eigenvectors are shown in Figure 2.5. Due to the one of eigenvectorshas removed and left data that is only in terms of the other in Table 2.4

Table 2.3 Data Transformed with 2 Eigenvectors[33]

x	y
-.827970186	-.175115307
1.77758033	.142857227
-.992197494	.384374989
-.274210416	.130417207
-1.67580142	-.209498461
-.912949103	.175282444
.0991094375	-.349824698
1.14457216	.0464172582
.438046137	.0177646297
1.22382056	-.162675287

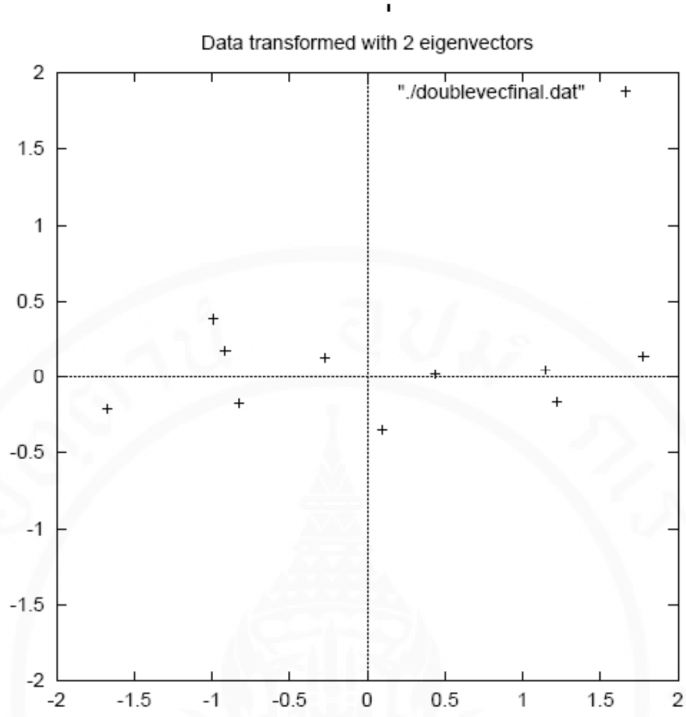


Figure 2.5 the Plots of New Data Point by Applying the PCA Analysis Using Both Eigenvectors [33]

Table 2.4 TheData after Transforming Using Only the Most Significant Eigenvector[33]

Transformed Data (Single eigenvector)
x
-.827970186
1.77758033
-.992197494
-.274210416
-1.67580142
-.912949103
.0991094375
1.14457216
.438046137
1.22382056

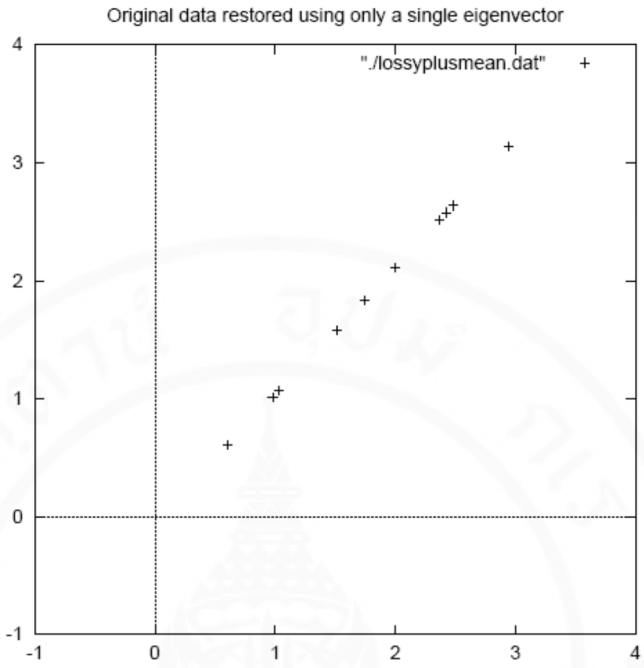
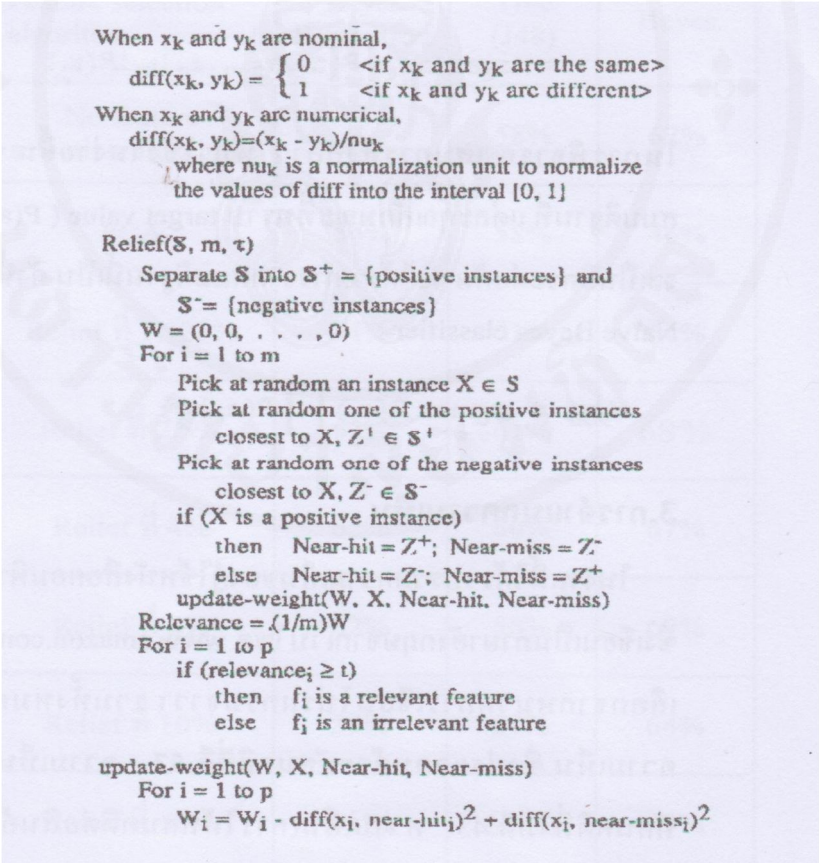


Figure 2.6 TheReconstruction from the Data that Was Derived Using Only ASingle Eigenvector[33]

Basically the data have been transformed, thus the data isexpressed in terms of the patterns between them as shown in Figure 2.6, where the patterns are the lines thatmost closely describe the relationships between the data. This is helpful because data point is classified as a combination of the contributions from each ofthose lines.Initially there are*x*and *y* axes. This is fine, but the values of each data point don't really tell us exactly how that point relates to the rest of the data. After gaining eigenvectors as axes, the values of the data points tell us exactly where (i.e. above/below) thetrend lines the data point sits. In the case of the transformation using botheigenvectors, the data is simply altered so that the data is in terms of those eigenvectors instead ofthe usual axes. Moreover, the single-eigenvector decomposition has removed the contribution due to the smaller eigenvalue and left data that is only in terms of the other.[33]

2.4.2 Relief Algorithm

For real-world problems involving much feature interaction, feature selection is need to be reliable and practically efficient to eliminate irrelevant features. The advantage of Relief is noise-tolerant and is not affected by feature interaction. The number of features which is performed on Relief is reduced due to the algorithm already selects the features which are relevant to the target concept and discards the features which are irrelevant to the target concept corresponding to the threshold. In addition, Relief uses the p-dimensional Euclid distance, which p is the number of features, for selecting Near-hit and Near-miss.[31]



When x_k and y_k are nominal,

$$diff(x_k, y_k) = \begin{cases} 0 & \text{<if } x_k \text{ and } y_k \text{ are the same>} \\ 1 & \text{<if } x_k \text{ and } y_k \text{ are different>} \end{cases}$$

When x_k and y_k are numerical,

$diff(x_k, y_k)=(x_k - y_k)/nu_k$

where nu_k is a normalization unit to normalize the values of diff into the interval [0, 1]

```
Relief(S, m, τ)
    Separate S into S+ = {positive instances} and
        S- = {negative instances}
    W = (0, 0, . . . , 0)
    For i = 1 to m
        Pick at random an instance X ∈ S
        Pick at random one of the positive instances
            closest to X, Z+ ∈ S+
        Pick at random one of the negative instances
            closest to X, Z- ∈ S-
        if (X is a positive instance)
            then   Near-hit = Z+; Near-miss = Z-
            else   Near-hit = Z-; Near-miss = Z+
        update-weight(W, X, Near-hit, Near-miss)
    Relevance = (1/m)W
    For i = 1 to p
        if (relevance_i ≥ t)
            then   f_i is a relevant feature
            else   f_i is an irrelevant feature

update-weight(W, X, Near-hit, Near-miss)
    For i = 1 to p
        W_i = W_i - diff(x_i, near-hit_i)^2 + diff(x_i, near-miss_i)^2
```

Figure 2.7 Relief Algorithm [31]

The process of Relief Algorithm is described by these followings and pseudo code for Relief Algorithm shown as Figure 2.7:

1. All samples are separated into positive or negative instances.
2. The weight of each feature initially is zero.
3. The algorithm picks one from all the samples.
4. The algorithm picks one of positive instances and one of negative instances.
5. The picked sample is identified as positive or negative one. If the picked sample is positive, Near-hit is the picked one from set of positive instances and the Near-miss is the picked one from set of negative instances. On the other hand, the picked sample is identified as positive or negative one. If the picked sample is negative, Near-hit is the picked one from set of negative instances and the Near-miss is the picked one from set of positive instances.
6. After the Near-hit and the Near-miss are obtained, the algorithm will update weight of all features. The weight of a feature will be subtracted by square of difference between the feature and the Near-hit because both of them are in the same class. Thus, difference value between them should not exist. Consequently, the weight of the feature will be added by square of difference between the feature and the Near-miss because both of them are in the different class. Thus, different value between them should exist.
7. The update feature weight vector is performed iteratively as the number of instances times.
8. After the process of (7) is finished, the feature weight vector is averaged by the number of instances.
9. Finally, the algorithm selects those features whose average weight, or relevance level, is above the given threshold.
10. The n features are obtained after the process of (9), which n is less than the number of the original features.

CHAPTER III
RESEARCH METHODOLOGY

3.1 Procedure of Research

The procedures of this research can divide into 4 main steps as collect-related information, steps of technique, test and evaluation, and conclusion and documentation. These steps are exhibited in Figure 3.1.

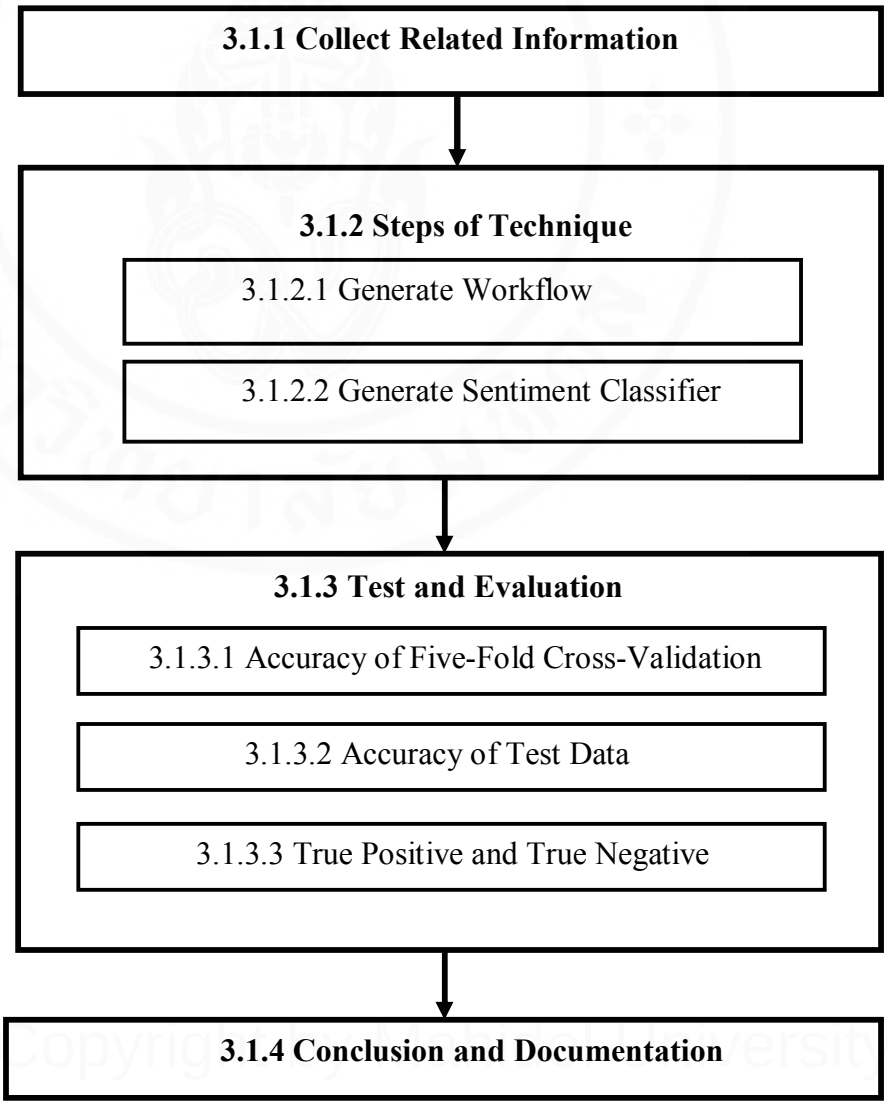


Figure 3.1 Procedures of Research

3.1.1 Collect Related Information

In this research, the secondary data that concern to sentiment classification are studied. These data compose of theories and related researches in

- sentiment classification,
- semantic-orientation approach to sentiment classification,
- machine-learning approach to sentiment classification,
- technique of using sentiment lexicon, i.e., SentiWordNet, and WordNet
- text pre-processing concept,
- concept of some machine learning techniques, i.e., Naïve Bayes, decision tree (J48), and Support Vector Machine (SVM)
- concept of some dimensional reduction techniques, i.e., Principal Component Analysis (PCA), and Relief Algorithm

The data are collected from textbooks and electronic data via Internet.

3.1.2 Steps of Technique

The technique of sentiment classification in this study involves 2 approaches which are the semantic-orientation approach, and the machine-learning approach. In addition, dimensional reduction is used to increase accuracy of classification.

In this study, there are three machine learning algorithms. Moreover, the proposed features, other than bag-of-words features, are used. Each learning algorithm executes on each type of features, then, compares accuracies of each type of features on that learning algorithm to find which features give highest accuracy on that algorithm. The effects of styles of feature sets on each learning algorithm are compared, and dimensional reduction techniques are applied to set of features too.

3.1.2.1 Generate Workflow

The workflow of the overall technique will be created in this step. This workflow integrates feature extraction which provides bag-of-words features and sentiment features for the machine-learning algorithms. Moreover, dimensional reduction techniques, i.e., Principal Component Analysis (PCA) and Relief Algorithm are integrated in this workflow. The sentiment-classification techniques are designed and generated to find the effects of the styles of feature sets, the two dimensional reduction

techniques, and the machine learning algorithms on improvement of the sentiment classification.

3.1.2.2 Generate sentiment classifier

In this step, the sentiment classifiers are developed. Overall components of the technique are these followings:

- The dataset is in English which is brought from website of Pang et al. [5]. The dataset is used to train a sentiment classifier.
- Text pre-processing is performed due to obtain data structure which is appropriate for the next step, feature extraction. The steps of text pre-processing which is used in this study, (1) part-of-speech tagging which the code for performing them are brought from Stanford Log-linear Part-Of-Speech Tagger (2) Elimination of Stop words (3) Stemming which the code for performing them are brought from Porter's algorithm
- Feature extraction is to obtain feature sets which are separated into 3 sets, (1) bag-of-words features (2) sentiment features (3) bag-of-words features in combination with sentiment features
- Dimensional reduction is performed, such as PCA. Moreover, algorithm which selects relevant features to target concept, such as Relief Algorithm is also used.
- Classification which the machine-learning algorithms, i.e., Naïve Bayes, decision tree (J48), and SVM, train sentiment classifiers with training dataset.

These components are created. Sentiment classifiers will be trained and tested their efficiency in the next step.

3.1.3 Test and Evaluation

The sentiment classifiers, which created in preceding step, are examined in their efficiency of classification. The sentiment classifiers that are trained on sentiment features and sentiment features in combination with bag-of-words features are compared to the sentiment classifier trained on bag-of-words features which are baseline in this study. Moreover, sentiment classifiers trained on set of features which

are brought from different dimensional reduction techniques are experimented which are expected to improve accuracy. This step can be divided as follows:

3.1.3.1 Accuracy of Five-Fold Cross-Validation

5-fold cross validation is used to evaluate the accuracy of sentiment model which is built during training time. The 5-fold cross-validation is performed initially by separating training dataset equally into 5 sets. In first round, the first set is kept for testing the model which is trained by the four remaining sets. Thus, the first accuracy of the sentiment model trained on the four remaining sets is obtained. In second round, the second set is kept for testing the model which is trained by four remaining sets. The second accuracy of classification is obtained. Furthermore, the third, fourth, fifth round is performed, as well as the first and second round, and the third, fourth, and fifth accuracies are obtained respectively. Then, an accuracy of sentiment model is averaged. This 5-fold cross-validation is to evaluate accuracy of sentiment model based on some training dataset, not unseen data. The 5-fold cross validation is shown in Figure 3.2

Set 1	Set 2	Set 3	Set 4	Set 5
test	train	train	train	train
train	test	train	train	train
train	train	test	train	train
train	train	train	test	train
train	train	train	train	test

Figure 3.2 Five-Fold Cross-Validation

3.1.3.2 Accuracy of test data

This test is performed due to validate the sentiment classifier by unseen data. The unseen data to test the sentiment classifier is test set which is provided initially by separating some of the dataset as the test data. Calculating accuracy of test data is shown in (3.1)

$$Accuracy\ of\ test\ data = \frac{number\ of\ correct\ output\ instances\ from\ test\ data\ set}{number\ of\ instances\ from\ test\ data\ set} * 100\ \% \quad (3.1)$$

3.1.3.3 True Positive and True Negative

True positive is the number which the result from the classification is positive and the actual result is also positive. On the other hand, True negative is the number which the result from the classification is negative and the actual result is also negative. Thus, the observers can determine whether performance of classification on negative or positive type is better and the other should be improved. The meanings of both true positive and true negative are also illustrated in Figure 3.3

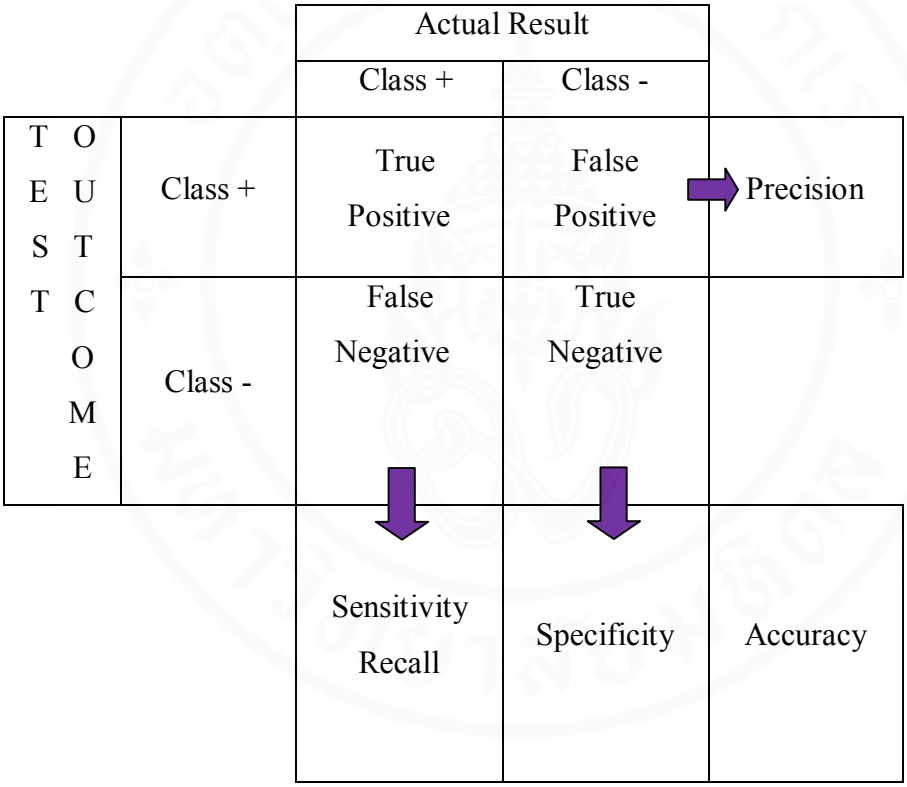


Figure 3.3 True Positive and True Negative

3.1.4 Conclusion and Documentation

The result of testing and study are analyzed, concluded, and presented with recommendations for future development. Finally, this study is documented.

3.2 Research Schedule

From the procedure of research can define research time shown in Table 3.1

Table 3.1 Research Time Table

Activities	Time							
	Month 1	Month 2	Month 3	Month 4	Month 5	Month 6	Month 7	Month 8
1.Collect Related Information	←	—	→					
2.Steps of Technique		←	—	—	—	→		
2.1 Generate Workflow		←	—	→				
2.2 Generate Sentiment Classifier			←	—	—	→		
3.Test and Evaluation					←	—	→	
3.1 Accuracy of 5-fold cross-validation					←	→		
3.2 Accuracy of Test Data						←	→	
3.3 True Positive and True Negative					←	—	→	
4.Conclusion and Documentation						←	—	→

3.3 Research Tools

Tools will be used in this study as follows:

Hardware:

- Notebook
  - CPU Intel^{R} Core^{TM}2 Duo processor T7300
  - RAM 2GB
  - Hard disk 160 GB

Software:

- OS
  - Windows Vista
- Development Tools
  - jdk1.6.0_24
  - editplus 3.31
  - Python 2.6.6
    - nltk-2.0b9.win32
    - numpy-MKL-1.6.0.win32-py2.6
- Data Mining Tool
  - Weka-3-6
- Natural Language Processing Tool
  - Stanford Log-linear Part-Of-Speech Tagger

CHAPTER IV
SENTIMENT CLASSIFICATION

This chapter presents an algorithm for sentiment classification. Sentiment features in combination with bag-of-words features are further constructed to improve accuracy in classification with some machine learning algorithms. In the beginning step, pre-processing of text is performed on raw data then,vectors of features are constructed, and the last, machine learning algorithms perform on training data, which are vectors of features, to generate sentiment classifiers. Three machine learning algorithms, i.e., SVM, Naïve Bayes, and Decision Tree are applied to sentiment classification in this study. In addition, dimension reduction techniques are used and expected to improve accuracy in classification.

The algorithm consists of five main parts,as

- pre-processing,
- semantic approach with SentiWordNet,
- feature extraction,
- dimensional reduction techniques, and
- machinelearning methods respectively.

Overall flowchart of the algorithms for sentiment classification is shown in Figure 4.1.

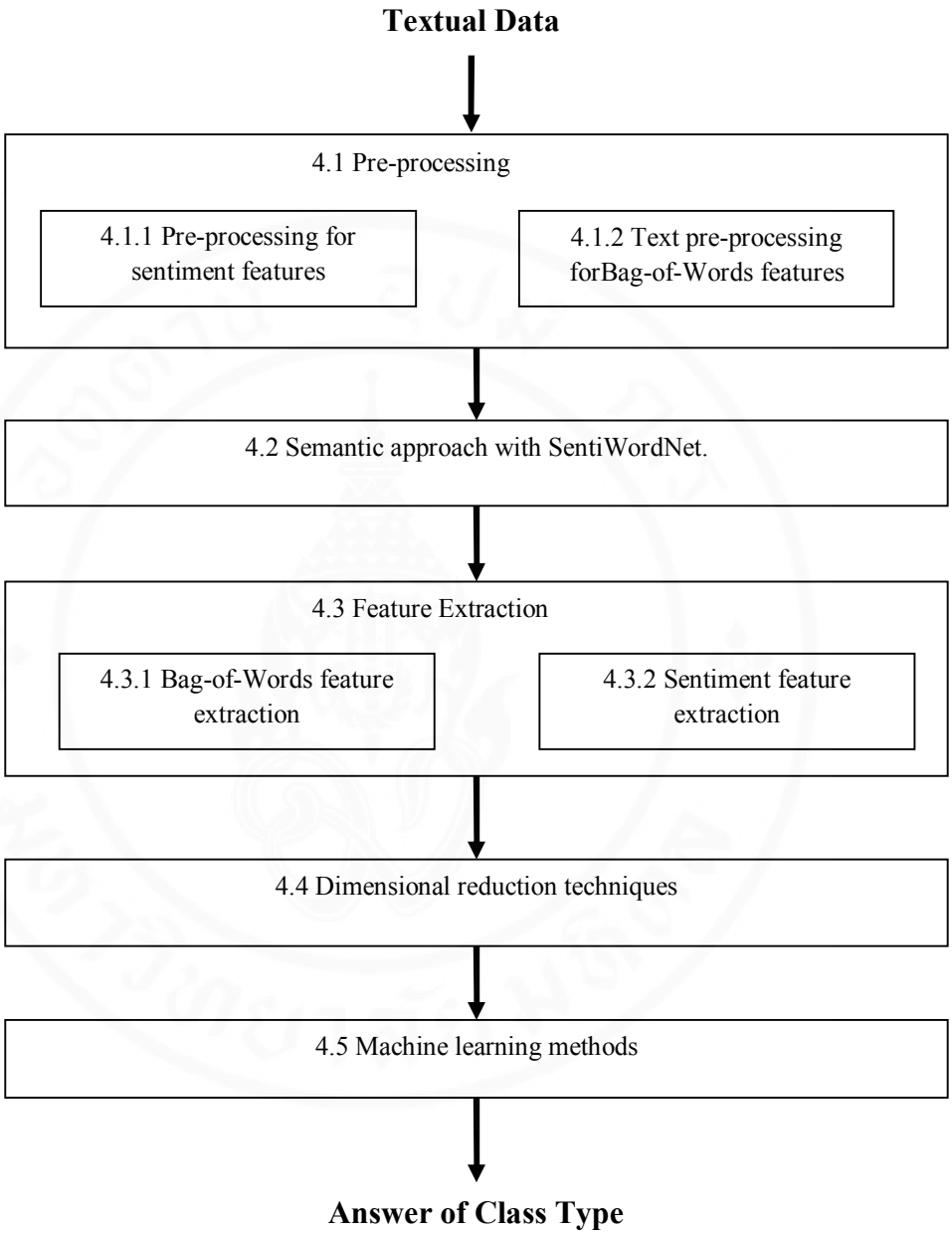


Figure 4.1 Algorithm Overview

4.1 Pre-processing

There are 2 methods of preprocessing which is used for sentiment features and bag-of-words features, respectively.

4.1.1 Pre-processing for sentiment features

The main objective of this pre-processing is to map several morphological forms of words to a common feature which can be found in the lexicon, SentiWordNet. The pre-processing is categorized according to concerned Part-of-speech(POS) of words in the document. The concerned POS are verb, adjective, adverb, and noun. The procedure is shown in Figure 4.2-4.6.

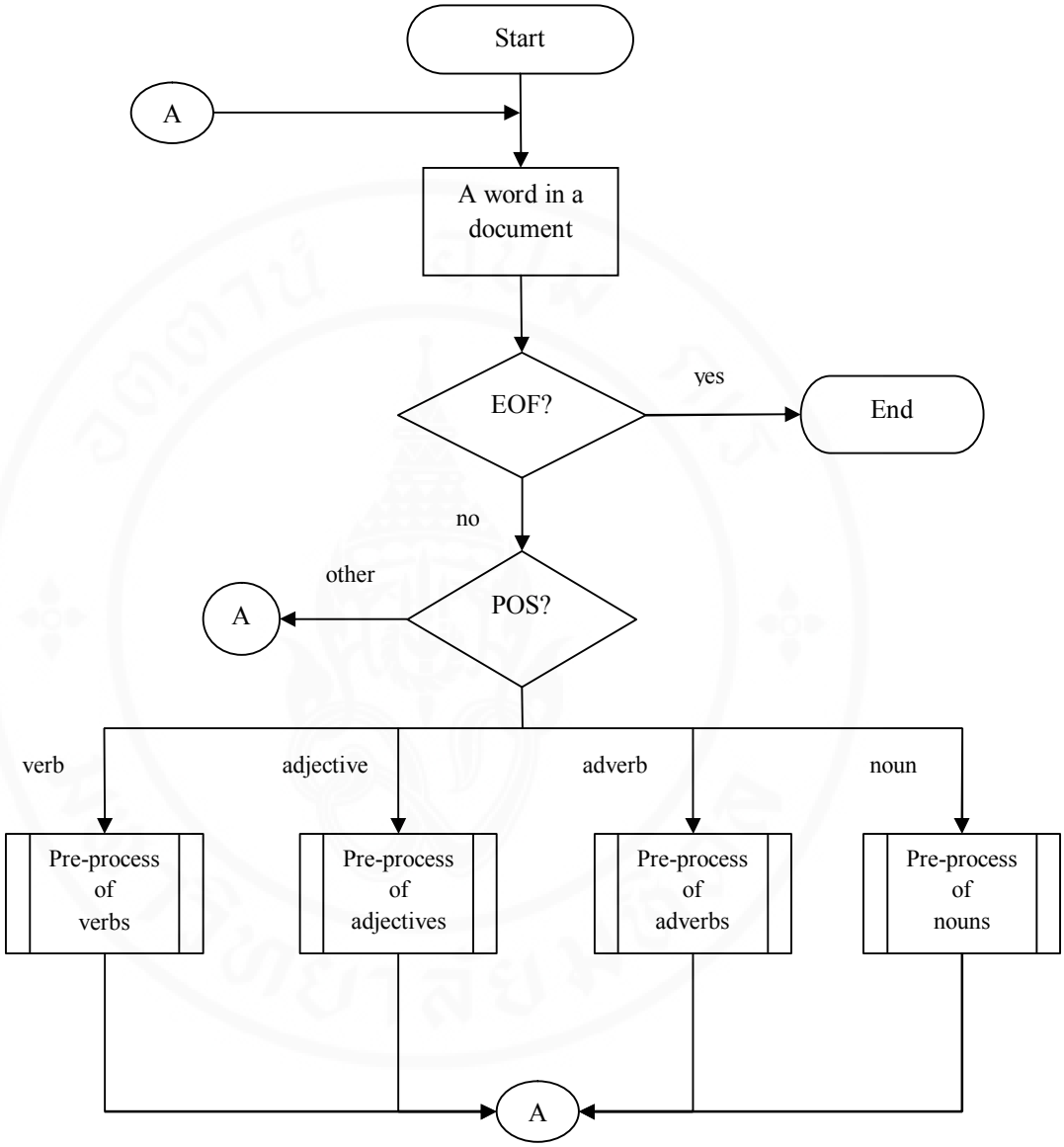


Figure 4.2 Pre-process of each concerned word in each document for next step of sentiment score assignment using SentiWordNet

The descriptions of Figure 4.2 are as these followings

1. To consider parts-of-speech (POS) of each word in the document, Stanford log-linear tagger and Penn Treebank Tagsetin appendix Aare used for tagging the words. If POS of word is verb, or adjective, or adverb, or noun, the word will be collected and transformedinto a standard form.

2. There are 4 different styles of pre-processing according to concerned POS. For example, a word with verb tagging will be performed by verb pre-processing. The other words are done by the pre-processing corresponding to their POS.

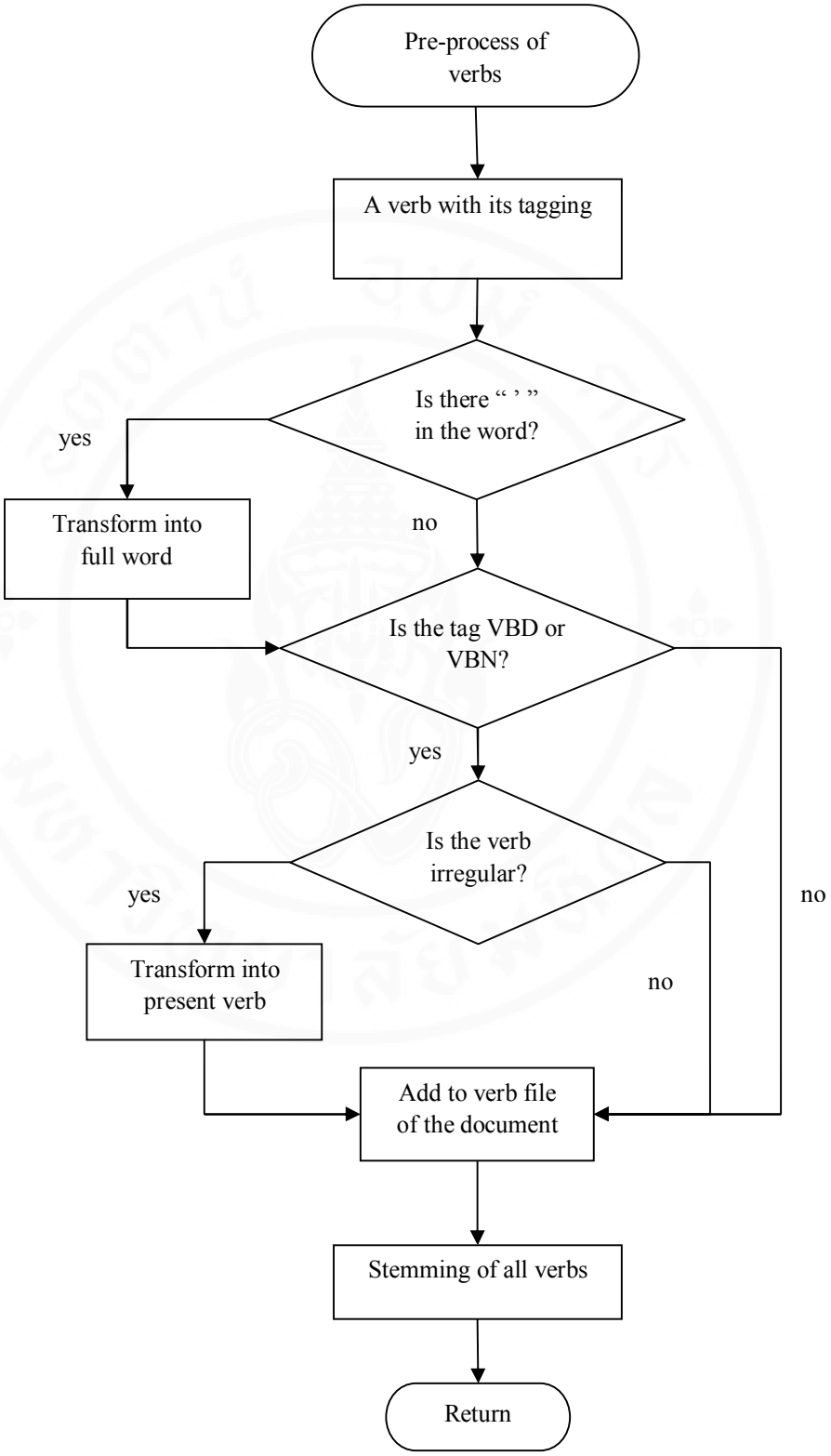


Figure 4.3 Details of pre-process for each verb

The descriptions of Figure 4.3 are as these followings

1. The various form of verb is related to sentence in which the verb is a component. Therefore, the verb is classified into VB, or VBD, or VBG, or VBN, or VBP, or VBZ.

2. The word which is in form of apostrophe will be transformed into full word.

3. The words with other tags, rather than VBD or VBN, the form of words will not be changed. Those words will be changed in the fifth step, stemming.

4. In the case of VBD orVBN, the following consideration is performed.

4.1 If the word isirregular form (such as slept and taken), the word will be transformed into normal form of present verb.

4.2 If the wordwith regular form(such as closed and dropped) will not be changed. Those words will be changed in the fifth step, stemming.

5. All verbs, bothof the transformed or not, arestemmed and gathered in the file of processed verb.

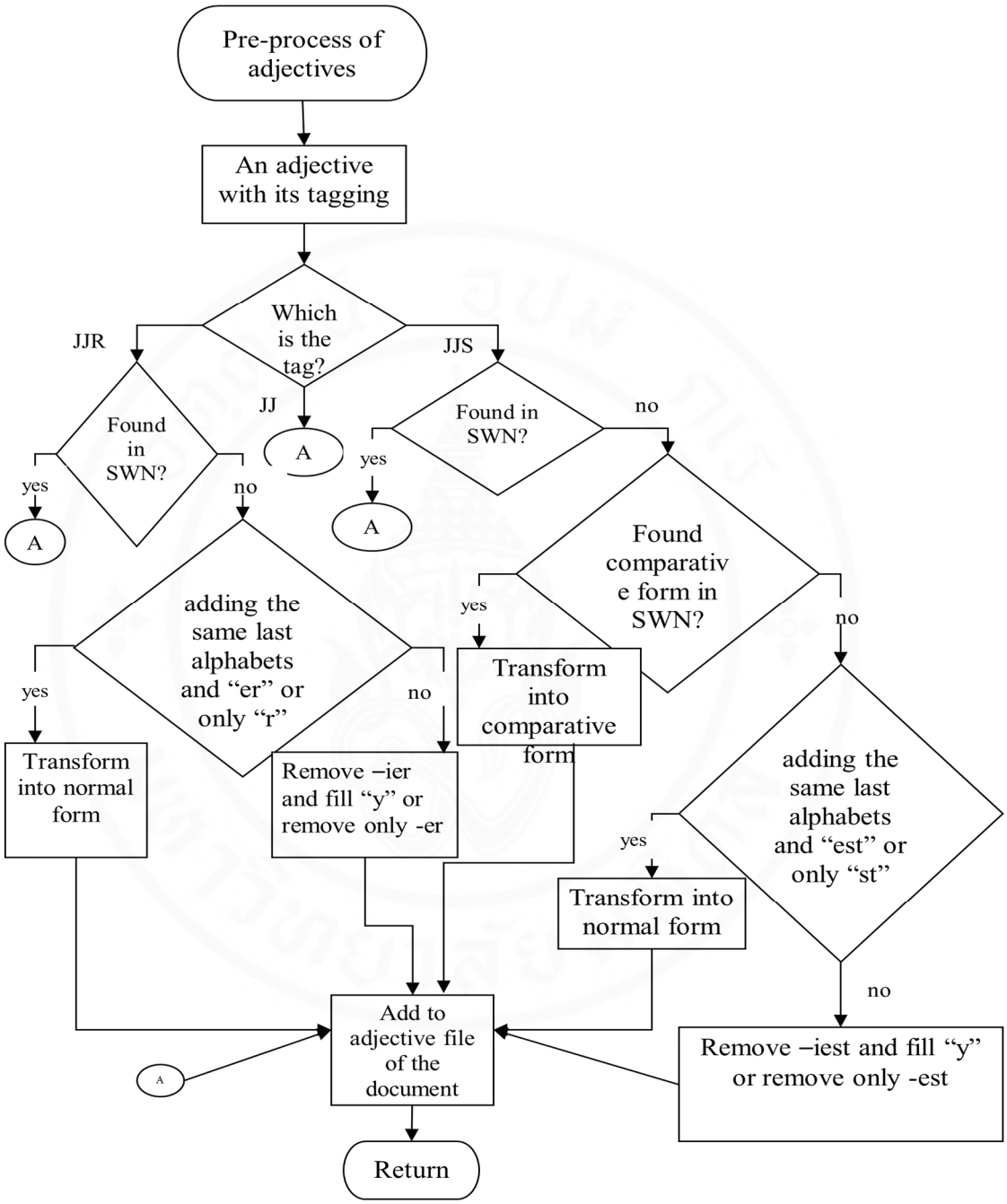


Figure 4.4 Details of pre-process for each adjective

The descriptions of Figure 4.4 are as these followings

1. The adjective is classified into JJ, or JJR, or JJS.
2. In the case of JJR, the following consideration is performed.

2.1 If the JJR can be found in SentiWordNet, the JJR form is used. Thus, the form of the word will not be changed.

2.2 In the case that the JJR cannot be found in SentiWordNet, the JJ form is expected to be found in SentiWordNet. Thus, the following consideration is performed.

2.2.1 If the JJR is come from (1) addingthe same last alphabet and "er" to its JJ (such as hotter) or (2) adding only "r" to its JJ (such as nicer), the JJR will be transformed into its root, JJ.

2.2.2 If the JJR is not involved in the case above, "ier" is removed and filled with "y" (such as easier) or removed "er" (such as colder) from JJR to form its root, JJ.

3. In the case of JJS, the following consideration is performed.

3.1 If the JJS can be found in SentiWordNet, the JJS form is used. Thus, the form of the word will not be changed.

3.2 If the JJS cannot be found inSentiWordNet, the JJR form is expected to be found in SentiWordNet. Thus, the following consideration is performed.

3.2.1 If the JJR for JJS can be found in SentiWordNet, the JJS will be transformed into its JJR, instead.

3.2.2 In the case that the JJR for JJS cannot be found in SentiWordNet, the JJ form is expected to be found in SentiWordNet. Thus, the following consideration is performed.

3.2.2.1 If the JJS is come from (1) adding the same last alphabet and "est" to its JJ (such as hottest) or (2) adding only "st" to its JJ (such as nicest), the JJS will be transformed into its root, JJ.

3.2.2.2 If the JJS is not involved in the case above, (1) "iest" is removed and filled with "y" (such as easiest) or removed "est" (such as coldest) from JJS to form its root, JJ.

4. The form of JJ will not be changed.

5. All adjectives, both of the transformed or not, are gathered in the file of processed adjective.

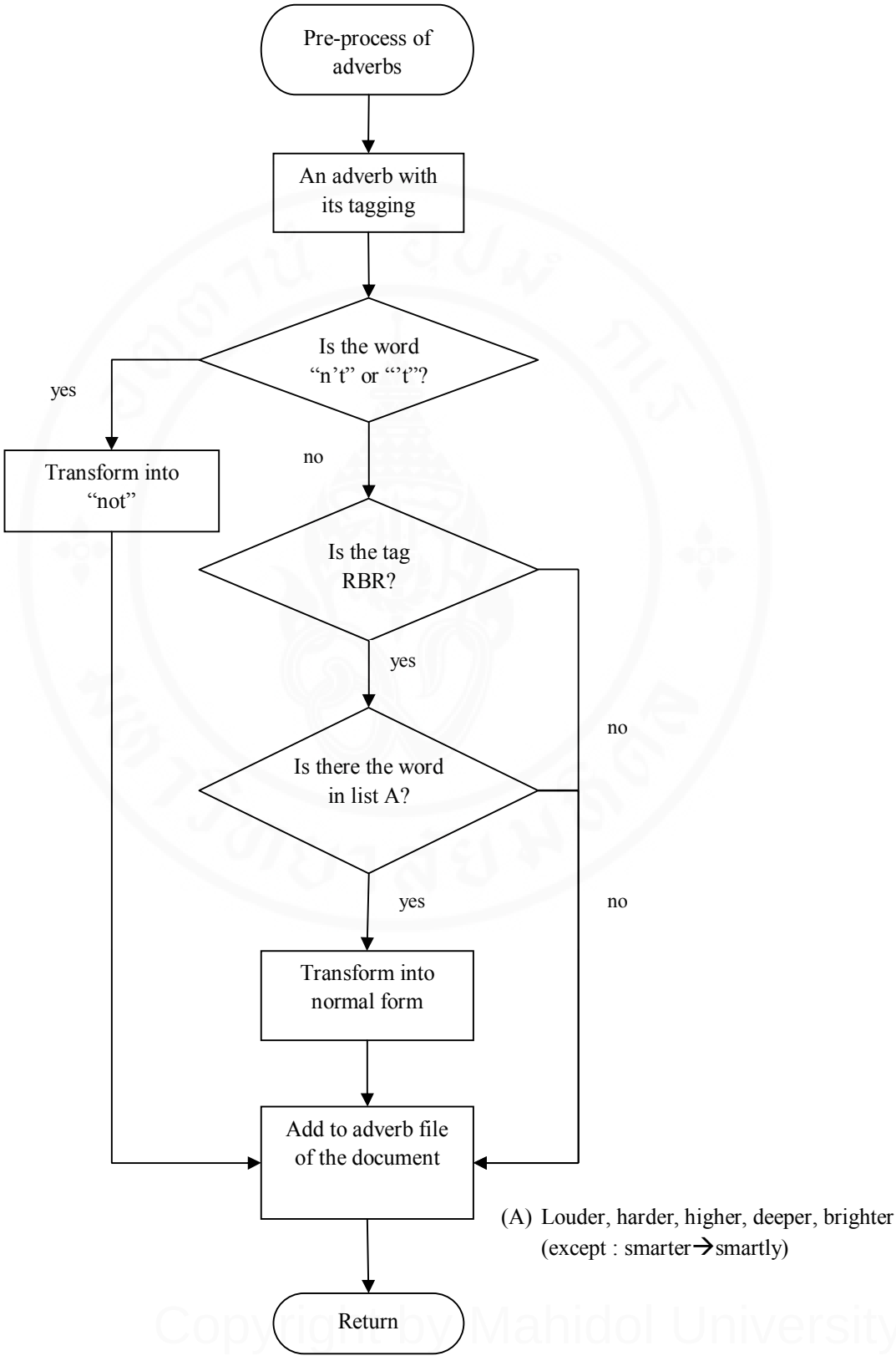


Figure 4.5 Details of pre-process for each adverb

The descriptions of Figure 4.5 are as these followings

1. The adverb is classified into RB, or RBR, or RBS.

2. The word which is an abbreviation of “not”, such as n’tor ‘t, will be transformed into full form, “not”.

3. In the case of RBR, the following consideration is performed.

3.1 If the RBR is in list (A), the RBR will be transformed into its RB. Exceptionally, if the word is “smarter”, the word will be transformed into “smartly”.

3.2 If the RBR is not in list (A), the form of RBR will not be changed.

4. The other forms of adverb, i.e. RB and RBS will not be changed.

5. All adverbs, both of the transformed or not, are gathered in the file of processed adverb.

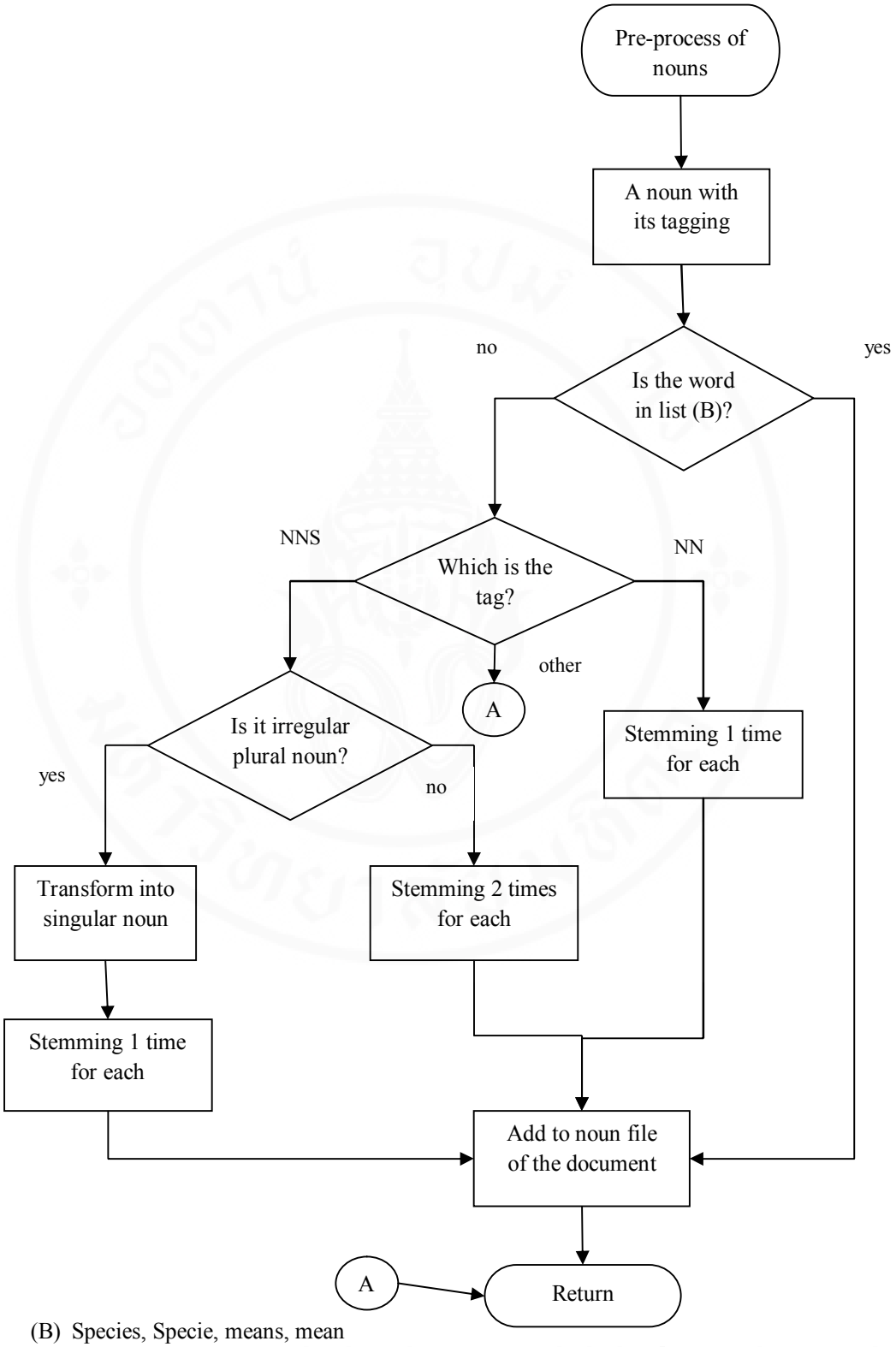


Figure 4.6 Details of pre-process for each noun

The descriptions of Figure 4.6 are as these followings

1. The noun is classified into NN, or NNS. The others, rather than these two tags, are disregarded because the other tags are expected not to be found in SentiWordNet.

2. In the case that the word is in the list (B), the word is collected immediately in the file of processed noun. The reason is that words in the list (B) are all singular nouns, although some words are like plural nouns. Thus, the words in the list (B) are certainly not to be changed.

3. In the case that the word is not in the list (B), the following consideration is performed.

3.1 If the word is NNS, the following consideration is performed.

3.1.1 If the NNS is irregular plural noun, the NNS will be transformed into its singular form in order that SentiWordNet contains only singular form of noun. After that, the stemming is performed on the transformed word one time.

3.1.2 If the NNS is regular plural noun, the stemming is performed on the word two times.

3.2 If the word is NN, the stemming is performed on the word one time.

4. All stemmed nouns are collected in file of processed noun.

4.1.2 Text Pre-Processing for Bag-of-Words Features

A textual document can be represented by bag-of-words features which are derived from a set of unique words in a document collection. In order to obtain all unique words, the text pre-processing is performed initially, i.e. tokenization, sentence splitting, Part-of-Speech tagging, elimination of stop words, and stemming. In addition, the first stemming is for transforming plural nouns into singular nouns. The second stemming is for transforming the words into their roots. The procedure is shown in Figure 4.7-4.9.

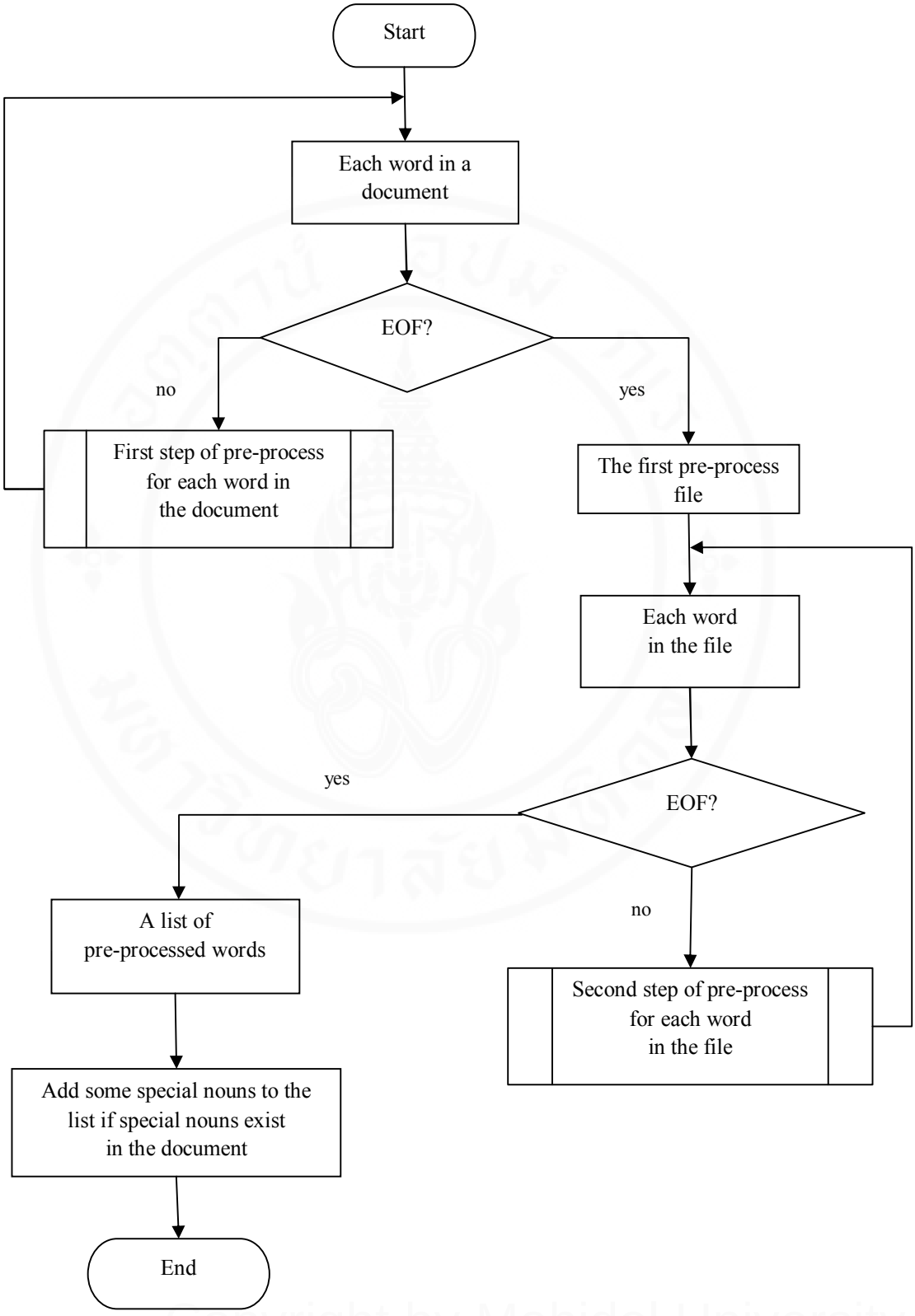


Figure 4.7 Pre-processing of each word in each document

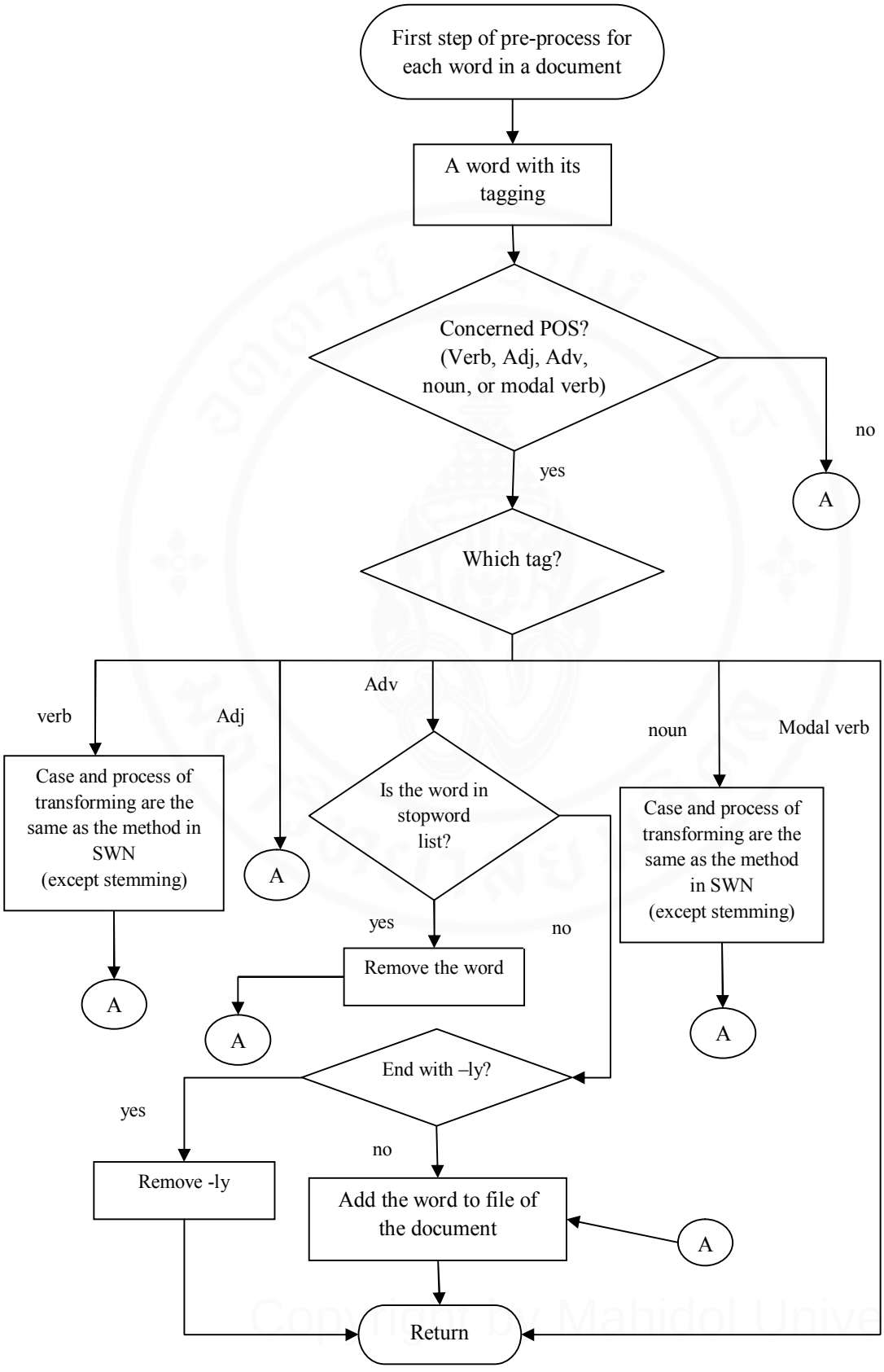


Figure 4.8 First step of pre-process

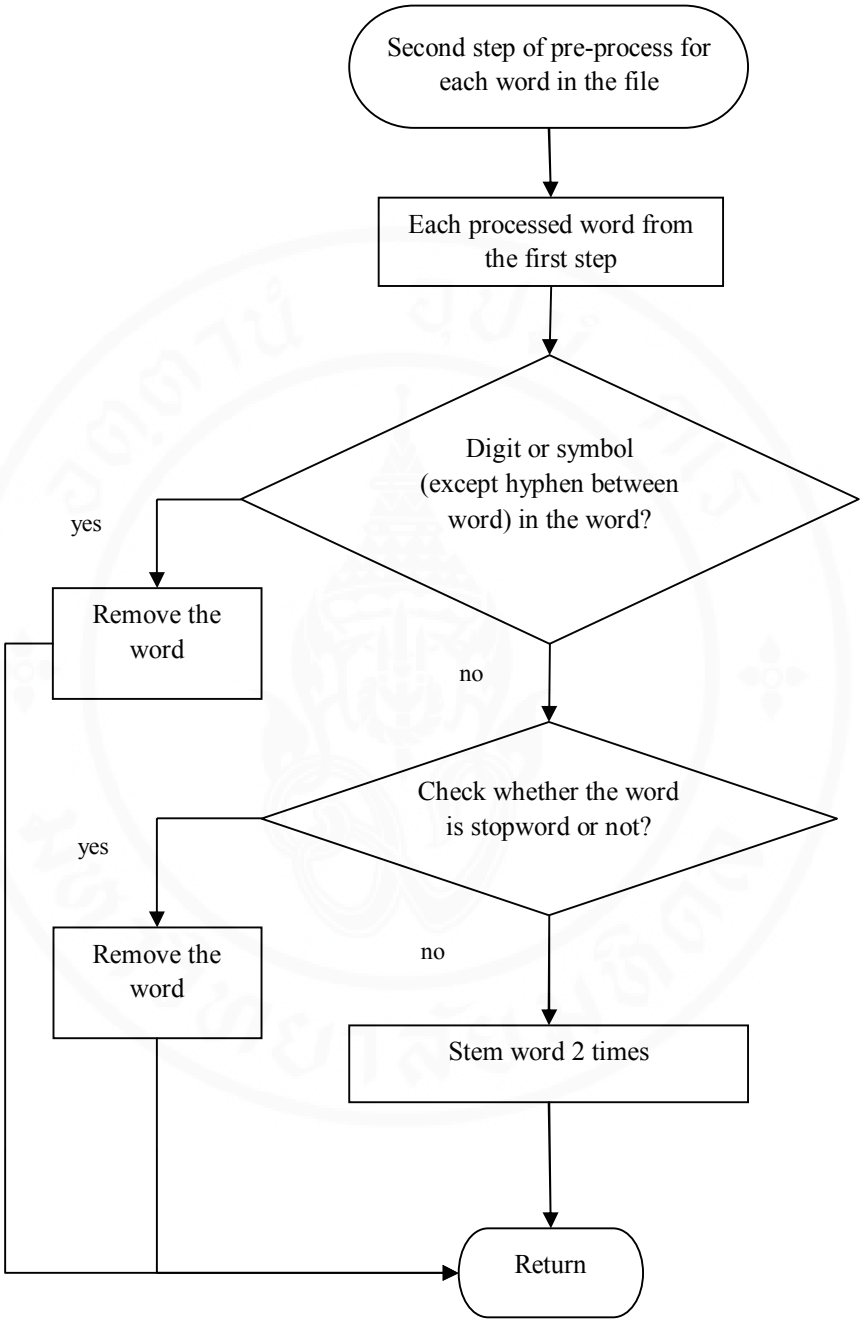


Figure 4.9 Second step of pre-process

The descriptions of Figure 4.7-4.9 are as these followings

1. If the program is still working on a word, the end of file is not met. The first step of pre-process for each word in the document is performed as followings.

1.1 The words which are selected for bag-of-words features are verbs, adjectives, adverbs, nouns, and modal verbs.

1.2 The first pre-process of a word is performed depending on its POS.

1.2.1 If a word is a verb, all pre-process is the same as the one for sentiment features but stemming is excluded. Stemming will be performed on verbs, together with other POS words.

1.2.2 If a word is an adjective, the form of the adjective is not changed.

1.2.3 If a word is an adverb, the following consideration is performed.

1.2.3.1 In the case that an adverb is in stopword list, the adverb will be removed.

1.2.3.2 In the case that an adverb is not in stopword list, the following consideration is performed.

-If the adverb ends with “ly”, “ly” is removed from the adverb to form an adjective, instead, in order to group into one adjective.

- If the adverb does not end with “ly”, the form of the adjective is not changed.

1.2.4 If a word is a noun, all pre-process is the same as the one for sentiment features but stemming is excluded. Stemming will be performed on nouns, together with other POS words.

1.2.5 If a word is a modal verb, the form of the modal verb is not changed.

1.3 All processed words from (1.1) and (1.2) are collected in a first processed file.

2. If the program ends, the end of file is met. The second step of pre-process for each processed word in the first pre-processed file is performed as followings.

2.1 Ifthe word has digit(s) or symbol(s), except hyphen between characters, the word is removed. Due to that, those words are considered to be low frequency words.

2.2 In the case that the word has no digit(s) or symbol(s), also the word has hyphen between characters, the following consideration is performed.

2.2.1 If the word is a stopword, the word will be removed.

2.2.2 If the word is not a stopword, the word is stemmed two times

2.3 All processed words from (2.1) and (2.2) are collected in a second pre-processed file, which is called as a list of pre-process words.

3. In addition, some special nouns which are similar to plural noun but the nouns are not plural and have specific meaning, i.e. species, specie, mean, means, are added to the list of pre-processed words if the words exists in any documents.

4.2 Semantic Approach with SentiWordNet

The sentiment scores for positivity, negativity, and objectivity of each word are derived from SentiWordNet. The concept of SentiWordNet is briefly described in section 4.2.1. The sentiment scores for a word with its specific POS are prepared by following equations 4.1-4.3 in section 4.2.2.

4.2.1SentiWordNet

SentiWordNet is a lexical resource in which each WordNetsynset is associated to three numerical scores i.e., objectivity, positivity, and negativity. The method which developed SentiWordNet is an adaptation of synset classification from previous work [34] [35] for deciding the positive-negative (PN) polarity and subjective-objective (SO) polarity of terms. Also, the method relies on training a set of ternary classifiers. A ternary classifier is a device which assigns the one of three labels to the input. Each of the classifiers is capable of deciding whether a synset is positive, or negative, or objective. Scores of each sentiment for a synset are determined by the normalized proportion of classifiers that have assigned the corresponding sentiment (or label) to the synset.

If all the classifiers agree in assigning the same label to a synset, that label will have the maximum score for that synset (value of 1), otherwise each label will have a score proportional to the number of classifiers that have assigned that label to the synset. Details of SentiWordNet will be described as followings:

Each ternary classifier is generated by using the semi-supervised method. A semi-supervised method is a learning process whereby only a small subset of the training data have been manually labeled while the remainder are instead unlabelled; processing by itself that has labeled the remainder automatically, by using training data with label as input.

To label some of unlabeled training data, two small sets (i.e., positive and negative synset) were manually selected as the intended synsets for 14 paradigmatic positive and negative terms. The positive and negative terms were used as seed words, thus positive and negative set of synsets are then iteratively expanded, in a number of iterations, into the final training sets. The expansion of two set used connection between synsets by WordNet lexical relations such as, “also-see” (the same-polarity)

and “direct antonymy” (the opposite PN-polarity). The label of the synset is based on the assumption is that synsets with similar gloss tend to have similar polarity. Furthermore, the objective synsets were defined as the set of synsets that do not belong to either positive or negative set, and contain terms not marked as either positive or negative in the General Inquirer lexicon, for any iteration of expansion.

Each synset was assigned a vectorial representation of gloss of the synset. Then, in training phase, vectorial representations of the training synsets for a given label are input to a standard ternary classifier. After training phase, the ternary classifiers were already generated to classify the remainder of synsets in WordNet. There are eight ternary classifiers which their training sets are different according to different values of iteration of expansion and/or different learners.

4.2.2 Preparing Sentiment Scores for A Word

SentiWordNet is a lexical resource, which used WordNet[36]synonym set. The synonym set is associated to three numeric sentiment scores, i.e., objective score, positive score, and negative score.The procedure and illustration to obtain sentiment score are as followings. Initially, the original SentiWordNet[37]is provided. In addition, SentiWordNet are categorized into four lists according to POS of synonym set, thus there are verb, adjective, adverb, and noun list of terms. As shown in Table 4.1-4.5, a synonym set contains terms which have the same meaning and corresponding POS.

Table 4.1 Verb Part of SentiWordNet

# POS	ID	PosScore	NegScore	SynsetTerms	Gloss
v	00062774	0.25	0	antisepticize#1	Disinfect with an antiseptic; "The animals were antisepticized by the veterinarian before the operation"
v	00062973	0	0	autoclave#1	subject to the action of an autoclave
v	00063095	0	0.125	hatch#1	emerge from the eggs; "young birds, fish, and reptiles hatch"
v	00063291	0.375	0.125	irritate#2	excite to an abnormal condition, or chafe or inflame; "Aspirin irritates my stomach"
v	00063557	0	0.75	inflame#5	become inflamed; get sore; "His throat inflamed"
v	00063724	0	0.875	inflame#1	cause inflammation in; "The repetitive motion inflamed her joint"
v	00063916	0.375	0.125	soothe#2	cause to feel better; "the medicine soothes the pain of the inflammation"

Table 4.1 Verb Part of SentiWordNet (cont.)

# POS	ID	PosScore	NegScore	SynsetTerms	Gloss
v	00064095	0.25	0	relieve#1 palliate#2 assuage#3 alleviate#1	provide physical relief, as from pain; "This pill will relieve your headaches"
v	00064487	0	0	massage#2	give a massage to; "She massaged his sore back"
v	00064643	0	0.5	hurt#2	give trouble or pain to; "This exercise will hurt your back"

Table 4.2 Adjective Part of SentiWordNet

# POS	ID	PosScore	NegScore	SynsetTerms	Gloss
a	00005718	0.125	0	infinite#4	total and all-embracing; "God's infinite wisdom"
a	00005839	0.5	0.125	living#3	(informal) absolute; "she is a living doll"; "scared the living daylights out of them"; "beat the living hell out of him"
a	00006032	0.25	0.5	relative#1 comparative#2	estimated by comparison; not absolute or complete; "a relative stranger"
a	00006245	0	0	relational#1	having a relation or being related
a	00006336	0	0	absorptive#1 absorbent#1	having power or capacity or tendency to absorb or soak up something (liquids or energy etc.); "as absorbent as a sponge"
a	00006777	0.375	0	sorbefacient#1 absorbefacient#1	inducing or promoting absorption
a	00006885	0	0.75	assimilatory#1 assimilative#2 assimilating#1	capable of taking (gas, light, or liquids) into a solution; "an assimilative substance

Table 4.2 Adjective Part of SentiWordNet (cont.)

# POS	ID	PosScore	NegScore	SynsetTerms	Gloss
a	00007096	0	0	hygroscopic#1	absorbing moisture (as from the air)
a	00007208	0	0.125	receptive#4	able to absorb liquid (not repellent); "the paper is ink-receptive"
a	00007331	0	0	shock-absorbent#1	having the capacity to absorb the energy of an impact; "the material absorbs shock and is used for shock-absorbent insoles"

Table 4.3 Adverb Part of SentiWordNet

# POS	ID	PosScore	NegScore	SynsetTerms	Gloss
r	00003771	0.625	0	blessedly#1	in a blessed manner
r	00003846	0	0.5	boiling#1	extremely; "boiling mad"
r	00003925	0.5	0.125	enviably#1	in an enviable manner; "she was enviably fluent in French"
r	00004038	0.125	0	pointedly#1	in such a manner as to make something clearly evident; "he pointedly ignored the question"
r	00004184	0	0.375	negatively#2	in a negative way; "he was negatively inclined"
r	00004288	0	0.75	negatively#1	in a harmful manner; "he was negatively affected"
r	00004394	0.5	0	kindly#1	in a kind manner or out of kindness; "He spoke kindly to the boy"; "she kindly overlooked the mistake"
r	00004567	0.625	0.125	unkindly#1	in an unkind manner or with unkindness; "The teacher treats the children unkindly"

Table 4.3 Adverb Part of SentiWordNet (cont.)

# POS	ID	PosScore	NegScore	SynsetTerms	Gloss
r	00004722	0	0	simply#1 only#1 merely#1 just#1 but#1	and nothing more; "I was merely asking"; "it is simply a matter of time"; "just a scratch"; "he was only a child"; "hopes that last but a moment"
r	00004967	0.75	0	simply#3	absolutely; altogether; really; "we are simply broke"

Table 4.4 Noun Part of SentiWordNet

# POS	ID	PosScore	NegScore	SynsetTerms	Gloss
n	00036580	0.5	0.125	cakewalk#2	an easy accomplishment; "winning the tournament was a cakewalk for him"; "invading Iraq won't be a cakewalk"
n	00036762	0.375	0.125	feat#1 exploit#1 effort#3	a notable achievement; "he performed a great feat"; "the book was her finest effort"
n	00037006	0.625	0	masterpiece#2	an outstanding achievement

Table 4.4 Noun Part of SentiWordNet (cont.)

# POS	ID	PosScore	NegScore	SynsetTerms	Gloss
n	00037090	0.75	0	masterstroke#1	an achievement demonstrating great skill or mastery
n	00037200	0.375	0.25	credit#4	used in the phrase `to your credit' in order to indicate an achievement deserving praise; "she already had several performances to her credit";
n	00037396	0	0	action#1	something done (usually as opposed to something said); "there were stories of murders and other unnatural actions"
n	00038175	0	0	res_gestae#2	things done
n	00038262	0	0	course_of_action#1 course#4	a mode of action; "if you persist in that course you will surely fail"; "once a nation is embarked on a course of action it becomes extremely difficult for any retraction to take place"

Table 4.4 Noun Part of SentiWordNet(cont.)

# POS	ID	PosScore	NegScore	SynsetTerms	Gloss
n	00038573	0	0.5	blind_alley#2	(figurative) a course of action that is unproductive and offers no hope of improvement; "all the clues led the police into blind alleys"; "so far every road that we've been down has turned out to be a blind alley"
n	00038863	0	0	collision_course#2	a course of action (following a given idea) that will lead to conflict if it continues unabated

From tables above, each synonym set has three sentiment scores. Each sentiment score describes the characteristics of objective, or positive, or negative of each term in the synonym set.

Furthermore, each term may have exists in many synonym sets or have several meaning. Therefore, each sentiment score for a term is the average score of all synonym sets in which the term exists and have the same POS.

To extract sentiment scores for each term in several synonym sets, all steps are shown as following:

1. Break group of terms in synonym set into many single terms with their orders, Furthermore, each term is determined as a key for its vector. The vector is used for keeping scores. For example, Thesecond synonym set which grateful exists is

found so that two scores (positive and negative) from synonym set of grateful#2 are second orderin grateful vector which are represented as “0.5,0.375”.

2. When any synonym set contains the term being concerned, two scores of that synonym set are kept in the term vector. Finally, when all synonym sets are searched whether the term exists, the vector of the term is already complete. For example, the vector of “grateful” has two members which can be shown as **vector of grateful=(“0.5,0.25”,“0.5,0.375”)**. In each member, the first number before “,” is positive score and the second one after “,”is negative score.

3. After that, all positive scores are averaged and all negative scores are averaged for the term. Score of objective is also calculated which equation is provided. Three equations is shown as followings 4.1-4.3:

These scores can be calculated by equation (4.1)-(4.3).

$$score_{pos}(A,o) = \frac{1}{n}\sum_{i=1}^{n} score_{pos,o}(i), \quad (4.1)$$

$$score_{neg}(A,o) = \frac{1}{n}\sum_{i=1}^{n} score_{neg,o}(i), \quad (4.2)$$

$$score_{obj}(A,o) = 1 - score_{pos}(A,o) - score_{neg}(A,o), \quad (4.3)$$

where$A$is a word with n corresponding synonym sets on a part of speech, and$score_{\{pos|neg|obj\},o}(i)$is the score of synonym sets $i$for term$A$at the part of speech$o$.

4. For equation (4.1)-(4.3), the word “grateful” has average of positive score $=1/2 * (0.5 + 0.5) = 0.5$ and average of score $=1/2 * (0.25 + 0.375) = 0.313$. The last one, average of objective score $= 1 - 0.5 - 0.313 = 0.187$

Conclusion for sentiment scores of “grateful” is

- Strength of positivity as 0.5,
- Strength of negativity as 0.313, and
- Strength of objectivity as 0.187.

Beside the word “grateful”, Table 4.5-4.8 show three average scores for each term.The POS list in this study means list of verb, or list of adjective, or list of

adverb, or list of noun extracted from SentiWordNetwhich each term accompanies with three sentiment scores.

Table 4.5 An Example of the Verb List with Their Sentiment Score

Stemmed verb	positivity	negativity	objectivity
brave	0.0	0.625	0.375
disarm	0.042	0.167	0.791
renov	0.208	0.0	0.792
effac	0.0	0.042	0.958
twang	0.025	0.175	0.8
zoom	0.0	0.0	1.0
jitterbug	0.0	0.0	1.0
twang	0.0	0.0	1.0
bravo	0.0	0.0	1.0
overbear	0.0	0.042	0.958

Table 4.6 An Example of the Adjective List with Their Sentiment Scores

adjective	positivity	negativity	objectivity
grateful	**0.5**	**0.313**	**0.187**
conditional	0.188	0.0	0.812
deferent	0.625	0.125	0.25
meatless	0.0	0.375	0.625
over-the-top	0.125	0.375	0.5
annexal	0.0	0.0	1.0
fairish	0.063	0.063	0.874
enjoyable	0.5	0.25	0.25
turbulent	0.0	0.375	0.625
favourable	0.563	0.094	0.343

Table 4.7 An Example of the Adverb List with Their Sentiment Scores

adverb	positivity	negativity	objectivity
reputedly	0.0	0.0	1.0
courageously	0.375	0.0	0.625
plaguey	0.0	0.25	0.75
provisionally	0.0	0.0	1.0
fleetly	0.25	0.0	0.75
oppositely	0.0	0.0	1.0
conservativey	0.125	0.0	0.875
unambitiousy	0.25	0.0	0.75
steady	0.25	0.0	0.75
sky-high	0.208	0.0	0.792

Table 4.8 An Example of the Noun List with Their Sentiment Scores

Stemmed noun	positivity	negativity	objectivity
dozens	0.0	0.25	0.75
eyeless	0.25	0.0	0.75
regiment	0.0	0.125	0.875
cuprimin	0.0	0.25	0.75
quiet	0.0	0.125	0.875
bradycardia	0.0	0.375	0.625
lasting	0.125	0.0	0.875
scalar	0.125	0.0	0.875
glare	0.042	0.042	0.916
organ	0.036	0.0	0.964

4.3 Feature Extraction

The representation of a review document is vectors of features which there are two distinctive types of features. The features are described by these followings.

4.3.1 Bag-of-Words Feature Extraction

This step follows the step in section 4.1.2, text pre-processing for bag-of-words features. There are files of pre-process stemmed words. All those words in those files are gathered and the redundancy of words which occur many times are eliminated to gain a set of unique words. Furthermore, the list in this section is referred to all unique terms in document collection. The first list of terms is already built. Due to many terms in the list, some terms are excluded from the list. The criterion for building the final complete list of terms is to exclude the terms which have more than nine characters. The number of unique terms in the list is 19,115. All terms in the list are called as "Bag-of-Words Features". The feature value is frequency of feature which occurs in the document.

The representation of the first review is shown as vector of bag-of-words features as following:

feature / document	author	work	win	……	better	fall	look
1	1	2	1		1	1	3

4.3.2 Sentiment Feature Extraction

This step follows the above step, preparing sentiment score for a word in section 4.2.2. A review document is described by sentiment features. Sentiment feature is applied by an average of eachtotal sentiment score from words in a POS.The sentiment features are used for four concerned POS (verb, adjective, adverb, and noun). In order to have three types of sentiment and four concerned POS, thus there are 12 features. A set of 12 sentiment features is {positivity of verb, negativity of

verb, objectivity of verb, positivity of adjective, negativity of adjective, objectivity of adjective, positivity of adverb, negativity of adverb, objectivity of adverb, positivity of noun, negativity of noun, objectivity of noun}. The value of each feature is real-number score as positivity, or negativity, or objectivity of the one POS.The features and their values are shown in Table 4.5

Table 4.9 An Example of Sentiment Features and Their Values

Sentiment type / Part of Speech	Positive	Negative	Objective
VERB	0.5	0.2	0.3
ADJECTIVE	0.7	0.2	0.1
ADVERB	0.4	0.3	0.3
NOUN	0.2	0.1	0.7

The Table 4.9 can be shown as a vector representation of a document, i.e. $\vec{d}$ = (0.5, 0.2, 0.3, 0.7, 0.2, 0.1, 0.4, 0.3, 0.3, 0.2, 0.1, 0.7)

The process and illustration to obtain sentiment features are as followings:

- The concerned POS files of a document are verb, adjective, adverb, and noun file. Each POS file of document is constructed by the steps and criteria in section 4.1.1. The file contains pre-processed wordsand all files are separated according to POS of the words in them.
- If a word in the concerned POS file of a document is also found in the corresponding POS list, three values of sentiment scores from the corresponding POS list are assigned to that word. For example, the word “enjoyable” which is in an adjective file of a document is found in the adjective list, thus the word obtains the sentiment scores, positive score as 0.5, negative score as 0.25, and objective score as 0.25.

- This process performs iteratively for each file of a document until all words in all files of the document are inspected. All words with specific POS which are found in corresponding POS list are given sentiment scores, positivity, negativity, and objectivity.

The examples of each file of a document with sentiment scores for each word are shown below:

davidmamet is a good director . mamet's an even better screenwriter and playwright . the guy's authored some of the best film and theatre works in the past decade -- the verdict , house of games , wag the dog , state and main , and the guy even won a pulitzer prize for his play glengarry glen ross . with that said , it's such a shame that his latest crime caper , heist , falls apart by employing too many of the well-known devices of a mamet production -- double-crossing femmes fatale , overtly memorable characters , and deceptive plot lines . but movies like the spanish prisoner , things change , and the winslow boy display a roundness to mamet's innate abilities . and it's almost a crime to witness how all of that goes awry in his latest film , heist . heist twists and turns along the road of documenting the shady life of career jewel thief joemoore (gene hackman) and his posse of thieves -- bobby blane (delroylindo) and don " pinky " pincus (ricky jay) . during a raid on a jewelry store , moore ends up with his mug on the surveillance cameras and he goes on the lam from the local marshals . he decides to get out of dodge with his recent bride fran (rebeccapidgeon , looking like a cross between sharon stone and joancrawford) , sailing into the sunset on his yacht . the only problem is that his fence bergman (dannydevito) has set up a big score -- the swiss job -- which he wants done so badly he holds joe's cut of the jewelry heist on layaway until the swiss job is done . to ensure his faithful hound brings the goods home intact , bergman tosses his nephew jimmy silk (samrockwell) into joe's crew for observation purposes . things become complicated when joe's wife has an affair with jimmy , the swiss job turns sour , devito starts acting like his character from ruthless people , and ricky jay starts looking for the exit to this mess of a film . amamet production usually derives its success from a solid emotional attachment to its characters -- usually they've been wronged and are seeking justice or absolution . the naïve scientist from the spanish prisoner or the curious psychologist from house of games were just decent people we recognize in our everyday lives . when those people get royally screwed over , its human nature to desire retribution . heist lacks any such emotional attachment . of greater concern is the recycling of plot points from ronin , which mamet also wrote and which deals with a similar plot involving double-crossing thieves on the run . to save the day , though , heist carries some of the best actors working today . devito ,lindo , and hackman chew up and spit

out the scenery around them . samrockwell -- who usually plays naïve and jovial characters in movies like galaxy quest -- plays silk as cold , manipulative , and downright spooky . but the main problem with heist is how surprisingly predictable the story is . unlike the similar yet far superior the score , there's never any surprise during heist's twisting . mamet has just become lazy with this one . everything about the film looks great , but there's nothing at all underneath . reviewed as part of our coverage of the 24th annual mill valley film festival .

Figure 4.10 The First Review in Data Collection

Table 4.10 Verb File of the First Review with Three Sentiment Scores

Stemmed verb	positivity	negativity	objectivity
author	0.0	0.0	1.0
act	0.075	0.0	0.925
bring	0.045	0.068	0.887
becom	0.031	0.0	0.969
becom	0.031	0.0	0.969
chang	0.05	0.038	0.912
complic	0.0	0.125	0.875
carri	0.019	0.006	0.975
chew	0.0	0.0	1.0
display	0.063	0.0	0.937
document	0.063	0.0	0.937
don	0.0	0.0	1.0
decid	0.0	0.0	1.0
dodg	0.0	0.0	1.0
do	0.01	0.038	0.952
do	0.01	0.038	0.952
deriv	0.0	0.0	1.0
desir	0.125	0.042	0.833
employ	0.0	0.0	1.0
end	0.0	0.0	1.0
ensur	0.0	0.0	1.0
fall	0.023	0.043	0.934
go	0.017	0.004	0.979
go	0.017	0.004	0.979
get	0.035	0.024	0.941
get	0.035	0.024	0.941
have	0.02	0.053	0.927

Table 4.10 Verb File of the First Review with Three Sentiment Scores (cont.)

Stemmed verb	positivity	negativity	objectivity
hold	0.059	0.01	0.931
have	0.02	0.053	0.927
have	0.02	0.053	0.927
have	0.02	0.053	0.927
involv	0.054	0.036	0.91
look	0.063	0.025	0.912
look	0.063	0.025	0.912
lack	0.125	0.0	0.875
look	0.063	0.025	0.912
play	0.039	0.007	0.954
recogn	0.167	0.0	0.833
review	0.0	0.0	1.0
say	0.011	0.0	0.989
sail	0.0	0.0	1.0
set	0.045	0.01	0.945
start	0.009	0.027	0.964
seek	0.05	0.0	0.95
screw	0.0	0.0	1.0
save	0.045	0.08	0.875
spit	0.0	0.125	0.875
toss	0.0	0.0	1.0
turn	0.014	0.014	0.972
twist	0.0	0.0	1.0
work	0.037	0.019	0.944
win	0.156	0.0	0.844
want	0.075	0.15	0.775

Table 4.10 Verb File of the First Review with Three Sentiment Scores (cont.)

Stemmed verb	positivity	negativity	objectivity
wrong	0.0	0.75	0.25
write	0.0	0.0	1.0
work	0.037	0.019	0.944

As mentioned above, sentiment features are focused on four of parts of speech which are verb, adjective, adverb, and noun. Therefore, Table 4.10 which shows verb part contains stemmed verbs and each verb has three sentiment scores which each sentiment score is derived from average score of all stemmed verbs with corresponding sentiment type. The equation is that following:

$$score\ of\ Verb\ POS_{senti} = \frac{1}{n} * \sum_{i=1}^{n} score\ of\ stemmed\ verb_{senti,i} \quad (4.4)$$

Wheresenti is sentiment type which can be positive, or negative, or objective, and n is number of stemmed verbs.

According to equation 4.4, values of three sentiments for verb POS are as followings:

Positivity of Verb POS=0.33

Negativity of Verb POS=0.036

Objectivity of Verb POS=0.932

Table 4.11 Adjective File of the First Review with Three Sentiment Scores

adjective	positivity	negativity	objectivity
annual	0.0	0.0	1.0
big	0.183	0.096	0.721
better	0.625	0.0	0.375
best	0.75	0.0	0.25
best	0.75	0.0	0.25
curious	0.042	0.125	0.833
cold	0.154	0.365	0.481
deceptive	0.063	0.688	0.249
decent	0.417	0.0	0.583
emotional	0.406	0.125	0.469
everyday	0.333	0.083	0.584
emotional	0.406	0.125	0.469
faithful	0.417	0.208	0.375
good	0.595	0.006	0.399
great	0.292	0.021	0.687
greater	0.5	0.25	0.25
human	0.0	0.0	1.0
innate	0.125	0.292	0.583
intact	0.25	0.0	0.75
jovial	0.75	0.125	0.125
local	0.0	0.0	1.0
lazy	0.0	0.0	1.0
latest	0.063	0.0	0.937
latest	0.063	0.0	0.937
many	0.0	0.0	1.0
memorable	0.25	0.125	0.625
manipulative	0.25	0.0	0.75

Table 4.11 Adjective File of the First Review with Three Sentiment Scores (cont.)

adjective	positivity	negativity	objectivity
main	0.208	0.125	0.667
only	0.0	0.0	1.0
past	0.0	0.125	0.875
predictable	0.0	0.0	1.0
recent	0.0	0.0	1.0
ruthless	0.125	0.875	0.0
such	0.0	0.125	0.875
spanish	0.0	0.0	1.0
shady	0.156	0.281	0.563
swiss	0.0	0.0	1.0
swiss	0.0	0.0	1.0
swiss	0.0	0.0	1.0
sour	0.042	0.354	0.604
solid	0.142	0.108	0.75
spanish	0.0	0.0	1.0
such	0.0	0.125	0.875
similar	0.225	0.125	0.65
spooky	0.0	0.375	0.625
similar	0.225	0.125	0.65
superior	0.321	0.0	0.679
well-known	0.063	0.063	0.874

Table 4.11 which shows adjective part contains adjective and each adjective has three sentiment scores. Each sentiment score is derived from average score of all adjectives with corresponding sentiment type. The equation is that following:

$$score\ of\ Adjective\ POS_{senti} = \frac{1}{n} * \sum_{i=1}^{n} score\ of\ Adjective_{senti,i} \quad (4.5)$$

Wheresenti is sentiment type which can be positive, or negative, or objective, and n is number of adjectives.

According to equation 4.5, values of three sentiments for Adjective POS are as followings:

Positivity of Adjective POS=0.191

Negativity of Adjective POS=0.113

Objectivity of Adjective POS=0.695

Table 4.12 Adverb File of the First Review with Three Sentiment Scores

adverb	positivity	negativity	objectivity
apart	0.0	0.083	0.917
almost	0.0	0.0	1.0
awry	0.188	0.0	0.812
also	0.0	0.0	1.0
as	0.0	0.125	0.875
badly	0.05	0.475	0.475
downright	0.75	0.0	0.25
even	0.125	0.125	0.75
even	0.125	0.125	0.75
far	0.025	0.0	0.975
just	0.104	0.0	0.896
just	0.104	0.0	0.896
never	0.063	0.438	0.499
overtly	0.5	0.125	0.375
royally	0.125	0.0	0.875
so	0.0	0.0	1.0
surprisingly	0.125	0.0	0.875
too	0.063	0.125	0.812

Table 4.12 Adverb File of the First Review with Three Sentiment Scores (cont.)

adverb	positivity	negativity	objectivity
though	0.0	0.0	1.0
usually	0.0	0.0	1.0
usually	0.0	0.0	1.0
up	0.1	0.025	0.875
usually	0.0	0.0	1.0
yet	0.104	0.208	0.688

Table 4.12 which shows adverb part contains adverb and each adverb has three sentiment scores. Each sentiment score is derived from average score of all adverbs with corresponding sentiment type. The equation is that following:

$$score\ of\ Adverb\ POS_{senti} = \frac{1}{n} * \sum_{i=1}^{n} score\ of\ Adverb_{senti,i} \quad (4.6)$$

Wheresenti is sentiment type which can be positive, or negative, or objective, and n is number of adverb.

According to equation 4.6, values of three sentiments for Adverb POS are as followings:

Positivity of Adverb POS=0.106

Negativity of Adverb POS=0.077

Objectivity of Adverb POS=0.816

Table 4.13 Noun File of the First Review with Three Sentiment Scores

Stemmed noun	positivity	negativity	objectivity
affair	0.25	0.0	0.75
attach	0.0	0.0	1.0
absolut	0.0	0.25	0.75
attach	0.0	0.0	1.0
abil	0.313	0.063	0.624
actor	0.0	0.0	1.0
boy	0.063	0.031	0.906
bride	0.0	0.0	1.0
bergman	0.0	0.0	1.0
bergman	0.0	0.0	1.0
crime	0.0	0.188	0.812
caper	0.104	0.104	0.792
crime	0.0	0.188	0.812
career	0.0	0.0	1.0
cross	0.0	0.0	1.0
crawford	0.0	0.0	1.0
cut	0.038	0.006	0.956
crew	0.0	0.0	1.0
charact	0.264	0.0	0.736
concern	0.2	0.2	0.6
coverag	0.0	0.083	0.917
charact	0.264	0.0	0.736
camera	0.0	0.0	1.0
charact	0.264	0.0	0.736
charact	0.264	0.0	0.736
director	0.0	0.0	1.0
decad	0.0	0.0	1.0
dog	0.0	0.161	0.839

Table 4.13 Noun File of the First Review with Three Sentiment Scores (cont.)

Stemmed noun	positivity	negativity	objectivity
day	0.0	0.0	1.0
devic	0.025	0.05	0.925
deal	0.0	0.0	1.0
exit	0.0	0.083	0.917
film	0.0	0.0	1.0
film	0.0	0.0	1.0
fenc	0.0	0.0	1.0
film	0.0	0.0	1.0
film	0.0	0.0	1.0
film	0.0	0.0	1.0
festiv	0.0	0.0	1.0
guy	0.0	0.0	1.0
guy	0.0	0.0	1.0
glengarri	0.0	0.0	1.0
gene	0.0	0.0	1.0
galaxi	0.0	0.0	1.0
game	0.0	0.0	1.0
good	0.813	0.0	0.187
game	0.0	0.0	1.0
glen	0.0	0.0	1.0
hous	0.0	0.0	1.0
heist	0.0	0.063	0.937
heist	0.0	0.063	0.937
heist	0.0	0.063	0.937
heist	0.0	0.063	0.937
hound	0.0	0.313	0.687
home	0.028	0.0	0.972
hous	0.0	0.0	1.0

Table 4.13 Noun File of the First Review with Three Sentiment Scores (cont.)

stemmed noun	positivity	negativity	objectivity
heist	0.0	0.063	0.937
heist	0.0	0.063	0.937
heist	0.0	0.063	0.937
heist	0.0	0.063	0.937
jewel	0.0	0.0	1.0
jay	0.0	0.0	1.0
jewelri	0.125	0.0	0.875
job	0.019	0.058	0.923
jewelri	0.125	0.0	0.875
job	0.019	0.058	0.923
jimmi	0.0	0.0	1.0
jimmi	0.0	0.0	1.0
job	0.019	0.058	0.923
jay	0.0	0.0	1.0
justic	0.125	0.063	0.812
life	0.018	0.0	0.982
lam	0.0	0.0	1.0
life	0.018	0.0	0.982
line	0.0	0.063	0.937
mamet	0.0	0.0	1.0
main	0.0	0.0	1.0
mamet	0.0	0.0	1.0
mamet	0.0	0.0	1.0
moor	0.0	0.0	1.0
mug	0.125	0.375	0.5
mess	0.0	0.104	0.896
mamet	0.0	0.0	1.0
mamet	0.0	0.0	1.0

Table 4.13 Noun File of the First Review with Three Sentiment Scores (cont.)

stemmed noun	positivity	negativity	objectivity
mamet	0.0	0.0	1.0
mill	0.0	0.0	1.0
movi	0.0	0.0	1.0
marshal	0.188	0.0	0.812
movi	0.0	0.0	1.0
nephew	0.0	0.0	1.0
natur	0.1	0.0	0.9
noth	0.125	0.125	0.75
observ	0.05	0.0	0.95
playwright	0.0	0.0	1.0
pulitz	0.0	0.0	1.0
prize	0.0	0.0	1.0
play	0.0	0.0	1.0
product	0.0	0.0	1.0
plot	0.063	0.0	0.937
prison	0.0	0.063	0.937
poss	0.125	0.0	0.875
pinki	0.0	0.125	0.875
pincu	0.0	0.0	1.0
problem	0.042	0.083	0.875
product	0.0	0.0	1.0
prison	0.0	0.063	0.937
psychologist	0.0	0.0	1.0
plot	0.063	0.0	0.937
plot	0.063	0.0	0.937
problem	0.042	0.083	0.875
part	0.0	0.0	1.0
peopl	0.0	0.0	1.0
peopl	0.0	0.0	1.0

Table 4.13 Noun File of the First Review with Three Sentiment Scores (cont.)

stemmed noun	positivity	negativity	objectivity
Peopl	0.0	0.0	1.0
point	0.048	0.005	0.947
play	0.0	0.0	1.0
quest	0.0	0.0	1.0
round	0.0	0.0	1.0
road	0.0	0.25	0.75
raid	0.0	0.063	0.937
rebecca	0.0	0.0	1.0
retribut	0.167	0.042	0.791
recycl	0.0	0.0	1.0
run	0.025	0.0	0.975
rockwel	0.125	0.0	0.875
ross	0.0	0.0	1.0
screenwrit	0.0	0.0	1.0
state	0.031	0.031	0.938
shame	0.0	0.625	0.375
store	0.0	0.0	1.0
surveil	0.0	0.0	1.0
stone	0.0	0.0	1.0
sunset	0.0	0.0	1.0
score	0.125	0.0	0.875
silk	0.0	0.0	1.0
success	0.0	0.0	1.0
scientist	0.0	0.0	1.0
sceneri	0.0	0.0	1.0
sam	0.0	0.0	1.0
stori	0.0	0.063	0.937
score	0.125	0.0	0.875

Table 4.13 Noun File of the First Review with Three Sentiment Scores (cont.)

stemmed noun	positivity	negativity	objectivity
surpris	0.125	0.208	0.667
start	0.016	0.031	0.953
theater	0.0	0.0	1.0
thief	0.0	0.0	1.0
today	0.063	0.0	0.937
thief	0.0	0.0	1.0
thief	0.0	0.0	1.0
thing	0.0	0.0	1.0
twist	0.0	0.0	1.0
turn	0.0	0.0	1.0
thing	0.0	0.0	1.0
verdict	0.0	0.0	1.0
valley	0.0	0.0	1.0
wag	0.063	0.0	0.937
winslow	0.0	0.0	1.0
wit	0.125	0.0	0.875
wife	0.0	0.0	1.0
yacht	0.0	0.0	1.0

Table 4.13 which shows noun part contains stemmed noun and each stemmed noun has three sentiment scores. Each sentiment score is derived from average score of all stemmed nouns with corresponding sentiment type. The equation is that following:

$$score\ of\ Noun\ POS_{senti} = \frac{1}{n} * \sum_{i=1}^{n} score\ of\ Noun_{senti,i} \quad (4.7)$$

Wheresenti is sentiment type which can be positive, or negative, or objective, and n is number of stemmed noun.

According to equation 4.7, values of three sentiments for Noun POS are as followings:

Positivity of Noun POS=0.034

Negativity of Noun POS=0.031

Objectivity of Noun POS=0.936

From values of sentiment features above, the representation of the first review is shown as vector of sentiment features as following:

features / document	Pos of verb	Neg of verb	Obj of verb	Pos of adj	Neg of adj	Obj of adj	Pos of adv	Neg of adv	Obj of adv	Pos of noun	Neg of noun	Obj of noun
1	0.033	0.036	0.932	0.191	0.113	0.695	0.106	0.077	0.816	0.034	0.031	0.936

Vector $\vec{d}$= (0.033, 0.036, 0.932, 0.191, 0.113, 0.695, 0.106, 0.077, 0.816, 0.034, 0.031, 0.936)

4.4 Dimensional Reduction Technique

Dimensional reduction technique, Relief Algorithm and PCA are used to select features. Furthermore, these twoDimensional reduction techniquesareexpectedto improve accuracy of classification. The input of Relief Algorithm and PCA is combination features. PCA constructs relation of feature structures and selects linear combination features**.** The Relief Algorithm ranks features according to relevance to the target concept, then, features are selected which are expected to be potential feature to classify the reviews.

4.5 Machine Learning Methods

The various types of features with the learning algorithms, i.e. Naïve Bayes (multinomial), decision tree (J48), and SVM to generate sentiment classifiers are compared their potential in classifying reviews. The linear kernel is used for SVM. Due to that, our features, i.e., sentiment, and bag-of-words features are expected to be good potential for further step, machine-learning method, the not complicated kernel "linear kernel" is sufficient for classifying the reviews. Furthermore, using linear kernel for SVM prevents the overfitting with training data which can occur by using complicated kernel. Naïve Baye (multinomial) is used probability and statistics. Decision Tree (J48) is used entropy to apply for selecting attributes.

The comparison among algorithms will be shown in Chapter 5 Experimental Result. Furthermore, combination features (bag-of-words and sentiment features), bag-of-words features, and sentiment features are compared among them with three machine learning algorithms.

CHAPTER V
RESULTS

The experimental results in this study are tested in three main parts based on test process present in Chapter III. The first part is accuracy estimation. The second part is classification performance evaluation. The third part isaccuracy on unseen data. Initially, data used in this study is introduced in section 5.1. Eventually, combination features are ranked by Relief Algorithm in section 5.6 for seeking potential features for sentiment classification.

5.1 Data

Movie reviews in English language are used as dataset which are polarity dataset v1.0brought from website: http://www.cs.cornell.edu/people/pabo/movie-review-data/ [5]. Some reviews are removed from the dataset for more suitability by Nathan Treloar. Each review is labeled with respect to their overall sentiment polarity which is positive or negative. The total is 1,386 reviews. Dataset is separated into two groups in this study. The one of two groups is training data for machine learning to generate a sentiment classifier. The other is test data which is unseen data for assessing generalization of the sentiment classifier. Furthermore, the instances which are trained sentiment classifier are labeled as 694 positive and labeled as 692 negative reviews. The detail of data used in this study is shown in Table 5.1.

Table 5.1 Groups of Data

Sentiment / Set	positive	negative	total
training	554	554	1108
test	140	138	278
total	694	692	1386

5.2 Lexical Analysis

In a document, the negativity and positivity of words which are derived from SentiWordNetare used to determine orientation of words by calculating the difference between positivity and negativity of some words in a document. The words used in this lexical analysis are the same ones used in extracting sentiment features which are categorized according to their parts of speech and obtained the negativity, positivity scores from SentiWordNet.

For considering the word orientation, if the strength of the word is below zero, the word has negative orientation. If the strength of the word is above zero, the word has positive orientation. If the strength of the word is zero, the word has neutral orientation. The strength of all concerned words are summed up and averaged. To classify a document, if the sign of average strength is minus, the document is assigned into negative class, and if the sign of average strength is plus, the document is assigned into positive class. In the case of the average strength of the document is zero, the result is neutral which is considered as wrong answer because the training dataset are labelled their classes only as positive or negative, not neutral.

5.3 Accuracy Estimation

This section is divided into two parts. The first part is for five-fold cross-validation. The second part is for statistical test.

5.3.1 Five-Fold Cross-Validation

Each machine learning algorithm on each type of features is estimated accuracy by five-fold cross validation. Moreover, lexical analysis in section 5.2 is also experimented. The accuracies are compared among all machine learning algorithms on various types of features and, also lexical analysis as shown in Table 5.1.

Table 5.2 Results of various types of features with machine learning algorithms by five-fold cross-validation

Learning algorithms / Types of features	SVM (linear)	Naïve Bayes (multinomial)	Decision Tree (J48, prune)
Lexical analysis	65.79%		
Bag-of-words features	83.12% (±1.30)	79.69% (±3.64)	64.53% (±3.60)
Sentiment features	69.68% (±4.45)	70.40% (±3.42)	64.00% (±3.86)
Combination (bag-of-words + sentiment features)	**84.21% (±0.45)**	**79.87% (±3.74)**	**65.25% (±5.80)**

From results above, lexical analysis has accuracy at 65.79% which is lower than linear SVM and Naïve Bayes (multinomial) run through three different types of features. The sentiment features has quite good enough accuracy and take short time for three learning algorithms to run through sentiment features. The combination features are the most suitable features for three learning algorithms to

improve accuracies in classification from using Bag-of-words features with these learning algorithms. The combination features on linear SVM, Naïve Baye (multinomial) and Decision Tree (J48, prune) give accuracy 84.21%, 79.87%, and 65.25%, respectively. Thus, combination features on linear SVM obtain highest accuracy among three machine learning algorithms, which the accuracy is 84.21%.

For each type of features, the results are as followings:

- Bag-of-words features gain highest accuracy on SVM (linear)
- Combination features gain highest accuracy on SVM (linear)
- Sentiment features gain highest accuracy on Naïve Bayes (multinomial) which is slightly higher than the features on SVM (linear).

In conclusion, SVM (linear) is the most potential learning algorithm with various types of features because the almost accuracies are above 80%.

Decision Tree (J48, prune) has the least potential for classifying reviews with three sets of features in this study and all accuracies are between 64-66%

Naïve Bayes (multinomial) has good potential for classifying reviews with three sets of features in this study and the highest accuracy is 79.87% while the lowest accuracy is 70.40%. Moreover, Naïve Bayes (multinomial) is the most suitable learning algorithm which performs on sentiment features to gain the highest accuracy among three learning algorithms.

5.3.2 Statistical Test

The result of SVM with combination features istested with statistical method, one-tailed hypothesis, to evaluate the confidential level which combination features has more accuracy than bag-of-words features.

The higher accuracy of SVM with combination features than SVM with bag-of-words features

In sampling group size n, proportion of experimental success is $\hat{p}$. If n is very high value, distribution of $\hat{p}$ is nearly normal distribution, where $\mu_{\hat{p}} = p$, and $\sigma_{\hat{p}} = \sqrt{\frac{pq}{n}}$; q=1-p. Therefore, the equation is $Z = \frac{\hat{p} - p}{\sqrt{\frac{pq}{n}}}$.

Let $\hat{p}$= proportion of correct answers from SVM with combination features

P= proportion of correct answers from SVM with bag-of-words features

Hypothesis $H_0 : \hat{p} = p$

$H_1 : \hat{p} > p$

Assign n = 1108

$\hat{p}$= 0.8421

P = 0.8312

Formula $Z = \frac{\hat{p} - p}{\sqrt{\frac{pq}{n}}}$

$Z = \frac{0.8421 - 0.8321}{\sqrt{(0.8321)(0.1688)/1108}}$

= 0.965

Conclusion:Z has value at least 0.965 so that Z is in critical region. Therefore, H_0 is rejected, andH_1 is accepted. Due to that, Z cover 0.8328 of area under the curve, SVM with combination features has higher accuracy than SVM with bag-of-words features at the confidence level 83.28% by one-tailed test.

5.4 Classification Performance Evaluation

This section is divided into two parts. The first part is for recall, precision, and F-measure. The second part is for statistical test.

5.4.1Recall, Precision, and F-measure

Recall measures how well the sentiment classifier is doing at classifying all the relevant documents for concerned answers, which are positive reviews or negative reviews. Recall is defined [38]as

$$\text{Recall} = \frac{number\ of\ documents\ retrieved\ that\ are\ relavant}{total\ number\ of\ documents\ that\ are\ relavant} \quad (5.1)$$

Precisionmeasures how well the sentiment classifier is doing at rejecting non-relevant reviews. For example, the positive reviews are expected to be relevant reviews thus, the negative reviews are non-relevant reviews. Precision is defined [38] as

$$\text{Precision} = \frac{number\ of\ documents\ retrieved\ that\ are\ relavant}{total\ number\ of\ documents\ that\ are\ retrived} \quad (5.2)$$

F-measureis an effectiveness measure based on recall and precision that is used for evaluating classification performance and also in some search applications. It has the advantage of summarizing effectiveness in a single number. It is defined as the harmonic mean of recall and precision[39], which is

$$F = \frac{1}{\frac{1}{2}\left(\frac{1}{recall}+\frac{1}{precision}\right)} = \frac{2\times recall\times precision}{(recall+precision)} \quad (5.3)$$

Table 5.3 Recall, precision, and F-measure

Learning algorithms	Type of features	Class	Recall	Precision	F-measure
SVM (linear)	Bag-of-words	-	0.857	0.815	0.836
	sentiment		0.69	0.70	0.695
	combination		0.863	0.828	**0.845**
	Bag-of-words	+	0.805	0.850	0.827
	sentiment		0.704	0.694	0.699
	combination		0.821	0.857	**0.839**
Naïve Baye (multinomial)	Bag-of-words	-	0.827	0.780	0.803
	sentiment		0.688	0.711	0.699
	combination		0.838	0.777	**0.806**
	Bag-of-words	+	0.767	0.816	**0.791**
	sentiment		0.720	0.698	0.709
	combination		0.760	0.824	**0.791**
Decision Tree (J48)	Bag-of-words	-	0.641	0.647	0.644
	sentiment		0.664	0.633	0.648
	combination		0.673	0.646	**0.660**
	Bag-of-words	+	0.650	0.644	**0.647**
	sentiment		0.616	0.647	0.631
	combination		0.632	0.659	0.645

From Table 5.3, the conclusion is as following:

1. The experiment of SVM on combination features has F-measure at 0.845 and 0.839 for classification of negative class and positive class, respectively. Two values of F-measure from SVM on combination features are better than F-measure from SVM on bag-of-words features, which are 0.836 and 0.827 respectively. Therefore, classification by SVM on combination features is better than the one on

bag-of-words features. Furthermore, performance of classification for negative class is better than classification for positive class.

2. The experiment of Naïve Baye (multinomial) on combination features has F-measure at 0.806 and 0.791 for classification of negative class and positive class, respectively. F-measure of negative classification from Naïve Baye (multinomial) on combination features is slightly better than F-measure from Naïve Baye (multinomial) on bag-of-words features, which is 0.803. F-measure of positive classification from Naïve Baye (multinomial) on combination features is equal as the one on bag-of-words features, which is 0.791. Therefore, performance of classification by Naïve Baye (multinomial) on combination features is not clearly different from classification by the one on bag-of-words features.

3. The experiment of Decision Tree (J48) on combination features has F-measure at 0.660 and 0.645 for classification of negative class and positive class, respectively. F-measure of negative classification from Decision Tree (J48)on combination features is better than F-measure from Decision Tree (J48) on bag-of-words features, which is 0.644. F-measure of positive classification from Decision Tree (J48) on combination features is slightly lower than the one on bag-of-words features, which is 0.647. Therefore, performance of negative classification for Decision Tree (J48) on combination features is better than the one on bag-of-words features. On the other hand, performance of positive classification for Decision Tree (J48) on combination features is slightly worse than the one on bag-of-words features.

F-measure of SVM on combination features from both classificationimproves from the one on bag-of-words features, thus review classification by SVM on combination features has better performance than the one on bag-of-words features.

F-measure of Naïve Baye (multinomial) on combination features from both classificationdoes not improve clearly from the one on bag-of-words features, thus review classification by Naïve Baye (multinomial) on combination features has nearly performance as the one on bag-of-words features.

F-measure of Decision Tree (J48) on combination features for negative classification is higher than F-measure of Decision Tree (J48) on bag-of-words features. Therefore, performance of negative classification for Decision Tree (J48) on

combination features is better than the one on bag-of-words features. On the other hand, F-measure of Decision Tree (J48) on combination features for positive classification is slightly lower than F-measure of Decision Tree (J48) on bag-of-words features. Therefore, performance of positive classification for Decision Tree (J48) on combination features is worse than the one on bag-of-words features.

However, all learning algorithms on combination features has more value of F-measure for negative classification than F-measure for positive classification, thus performance for negative classification of all learning algorithms on combination features is better than performance for positive classification.

Two F-measure of SVM on combination features, for negative and positive classification, are the most values among F-measure of the other learning algorithms on any feature types on both negative and positive classification. Therefore, in this study, SVM (linear) on combination features has the best performance than the other learning algorithms on any feature types.

5.4.2 Statistical Test

The improvement of classification performance of SVM with combination features from SVM on bag-of-words features is tested statistic significant by one-tailed T-test at the confidential level 90%. This statistical test is divided into two parts. The difference of average F-measure of the negative classification is tested in the first part. The difference of average F-measure of the positive classification is tested in the second part.

The difference of average F-measure for negative classification

$$t = \frac{(\bar{X}_1 - \bar{X}_2) - (\mu_1 - \mu_2)}{S_p\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$$

$$\bar{X}_1 = 0.845, \quad S_1 = 0.003, \quad n_1 = 5$$
$$\bar{X}_2 = 0.836, \quad S_2 = 0.013, \quad n_2 = 5$$

$$S_p^2 = \frac{(n_1 - 1)S_1^2 + (n_2 - 1)S_2^2}{n_1 + n_2 - 2}$$

$$= \frac{4(0.003)^2 + 4(0.013)^2}{5 + 5 - 2}$$

$$S_p = 0.00943$$

Hypothesis $H_0: \mu_1 - \mu_2 = 0$

$H_1: \mu_1 - \mu_2 > 0$

Assign $\alpha = 0.1$

Critical region from table df = 5+5-2 = 8

$t_{0.1} = 1.397$

$$t = \frac{(0.845 - 0.836) - 0}{0.00943\sqrt{\frac{1}{5} + \frac{1}{5}}}$$

= 1.51

Conclusion $t_{compute} > t_{table}$, that means t is in critical region. Therefore, H_0 is rejected and H_1 is accepted. This shows that average F-measure of SVM with combination features is more than average F-measure of SVM on bag-of-words featuresfor negative classification at level of significant 0.1.

The difference of average F-measure for positive classification

$$t = \frac{(\bar{X}_1 - \bar{X}_2) - (\mu_1 - \mu_2)}{S_p\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$$

$\bar{X}_1 = 0.839, \quad S_1 = 0.00778, \quad n_1 = 5$

$\bar{X}_2 = 0.827, \quad S_2 = 0.014, \quad n_2 = 5$

$$S_p^2 = \frac{(n_1 - 1)S_1^2 + (n_2 - 1)S_2^2}{n_1 + n_2 - 2}$$

$$= \frac{4(0.00778)^2 + 4(0.014)^2}{5 + 5 - 2}$$

$$S_p = 0.011$$

Hypothesis $H_0: \mu_1 - \mu_2 = 0$

$H_1: \mu_1 - \mu_2 > 0$

Assign $\alpha = 0.1$

Critical region from table df = 5+5-2 = 8

$t_{0.1} = 1.397$

$$t = \frac{(0.839 - 0.827) - 0}{0.011\sqrt{\frac{1}{5} + \frac{1}{5}}}$$

= 1.724

Conclusion $t_{compute} > t_{table}$, that means t is in critical region. Therefore, H_0 is rejected and H_1 is accepted. This shows that average F-measure of SVM with combination features is more than average F-measure of SVM on bag-of-words features for positive classification at level of significant 0.1.

5.5 Accuracy on Unseen Data

Three learning algorithms on three types of features are estimated accuracies by five-fold cross-validation. The experiment which is SVM on combination features gives the highest accuracy, which is 84.21%. According to the result, the sentiment classifier which trained by SVM on combination features is chosen to use for further classifying new unseen movie reviews. Therefore, test data which is 278 reviews separated from data collection is used for testing the generalization of the sentiment classifier. As a result, the accuracy from the classifier is **85.25%**. Therefore, the sentiment classifier trained by SVM on combination features has generalization quality.

5.6 Dimension Reduction Technique

5.6.1 Relief Algorithm

Relief algorithm selects the features from a set of all features to gain the features which are relevant to target concept. In this study, selected features are expected to potentially classify reviews into positive or negative reviews. The combination features give the highest accuracy with each of three learning algorithms as shown in Table 5.2, thus Relief Algorithm is used for combination features. Relief Algorithm is used for the number of 100%, 90%, 80%, 70%, 60%, 50%, 40%, and 30% of all combination features. The various set of selected features with SVM give accuracy as shown in Table 5.4

Table 5.4 Selected combination features by Relief Algorithm with accuracies

Relief Algorithm at	Amount of selected feature	Accuracy
100%	19127	84.21%
90%	17214	80.69%
80%	15302	80.05%
70%	13389	80.32%
60%	11476	79.78%
50%	9564	79.42%
40%	7651	79.15%
30%	5738	78.34%

From the results above, selected features on SVM do not improve accuracy from all combination features on SVM

However, Relief Algorithm ranks the features according to relevant target concept. The first rank feature is **negativity of adjective**. The second rank feature is **positivity of adjective**. The third rank feature is **positivity of adverb**. Therefore, some of sentiment features in combination features have potential for sentiment classification.

The twenty top ranks of features are as followings:
negativity of adjective, positivity of adjective, positivity of adverb, well, also, homage, many, now, take, most, bad, inferior, work, in-crowd, re-write, re-inv, conspire, reshoot, place, maybe

5.6.2 Principal Component Analysis (PCA)

Although PCA is expected to apply to combination features, software cannot run PCA because of large amount of features. In study of Titima and Tanasanee[40], PCA is used for dimension reduction on 1666 features but the accuracies are not improved from using all features.

5.6.3 Correlation

The value of correlation indicates relationship between each feature and class. A positive relationship exists when frequency of positive feature or score of positivity of a POS increases, the class trends to be +1 or positive. If frequency of positive feature or score of positivity of a POS decreases, the class trends to be -1 or negative. Moreover, a positive relationship also exists when frequency of negative feature or score of negativity of a POS increases, the class trends to be -1 or negative. If frequency of negative feature or score of negativity of a POS decreases, the class trends to be +1 or positive.

A negative relationship exists when frequency of positive feature or score of positivity of a POS increases, the class trends to be -1 or negative. If frequency of positive feature or score of positivity of a POS decreases, the class trends to be +1 or positive. Moreover, a negative relationship also exists when frequency of negative feature or score of negativity of a POS increases, the class is probable to be +1 or positive. If frequency of negative feature or score of negativity of a POS decreases, the class trends to be -1 or negative. Therefore, if the absolute of correlation is high, capability of the feature to classify the reviews is also high.

Correlations between class and sentiment features are shown in Table 5.5. Correlations between class and bag-of-words features are shown in Table 5.6.

Table 5.5 Correlations between class and sentiment features

feature	rank fromRelief Algorithm	correlation between feature and class (descending and absolute)
Neg of adjective	1	-0.36
Pos of adjective	2	0.245
Neg of adverb	19115	-0.142
Obj of adjective	91	0.124
Pos of noun	174	0.114
Neg of verb	19125	-0.11
Pos of verb	19056	0.108
Pos of adverb	3	0.105
Obj of noun	19026	-0.093
Obj of adverb	18318	0.084
Neg of noun	19042	0.025
Obj of verb	18838	-0.00038

From Table 5.5, rank of featuresaccording to relevance to target concept from Relief Algorithm and correlation are not relative, although the first and second rank are related. However, rank of features from Relief Algorithm is more confidential than correlation because method of Relief Algorithm gives the weight of each feature from different value between one feature of an instance and one feature of the another instance. Also, the weight of each feature is updated many times. Then, the final weight of a feature comes up. Thus, Relief Algorithm gives the confidential weight of each feature in the vector. Consequently, the weight of features which is used to select features is also confidential.

Table 5.6 Correlation between class and bag-of-words features

feature	rank from Relief Algoritm	correlation between feature and class (descending and absolute)
bad	8	-0.24
also	2	0.186
most	7	0.17
mani	4	0.146
well	1	0.14
job	19	0.124
mayb	17	-0.109
homag	3	0.096
now	5	0.086
take	6	0.086
work	10	0.08
place	16	0.066
maguir	20	0.06
inferior	9	-0.035
in-crowd	11	-0.03
slam	18	0.016
conspir	14	0.01
reshoot	15	-2.78E-16
re-writ	12	-2.88E-17
re-inv	13	7.47E-19

From Table 5.6, rank of features according to relevance to target concept from Relief Algorithm and correlation are not relative. The weight of features from Relief Algorithm which is used to select features is also confidential which the reason is the same as from the Table 5.5.

CHAPTER VI
CONCLUSION AND RECOMMENDATION

In this chapter, there are three main parts, i.e., concept of work conclusion, conclusion of experimental result, and recommendation.

The semantic technique for sentiment classification can be obtain as shown in Figure 6.1

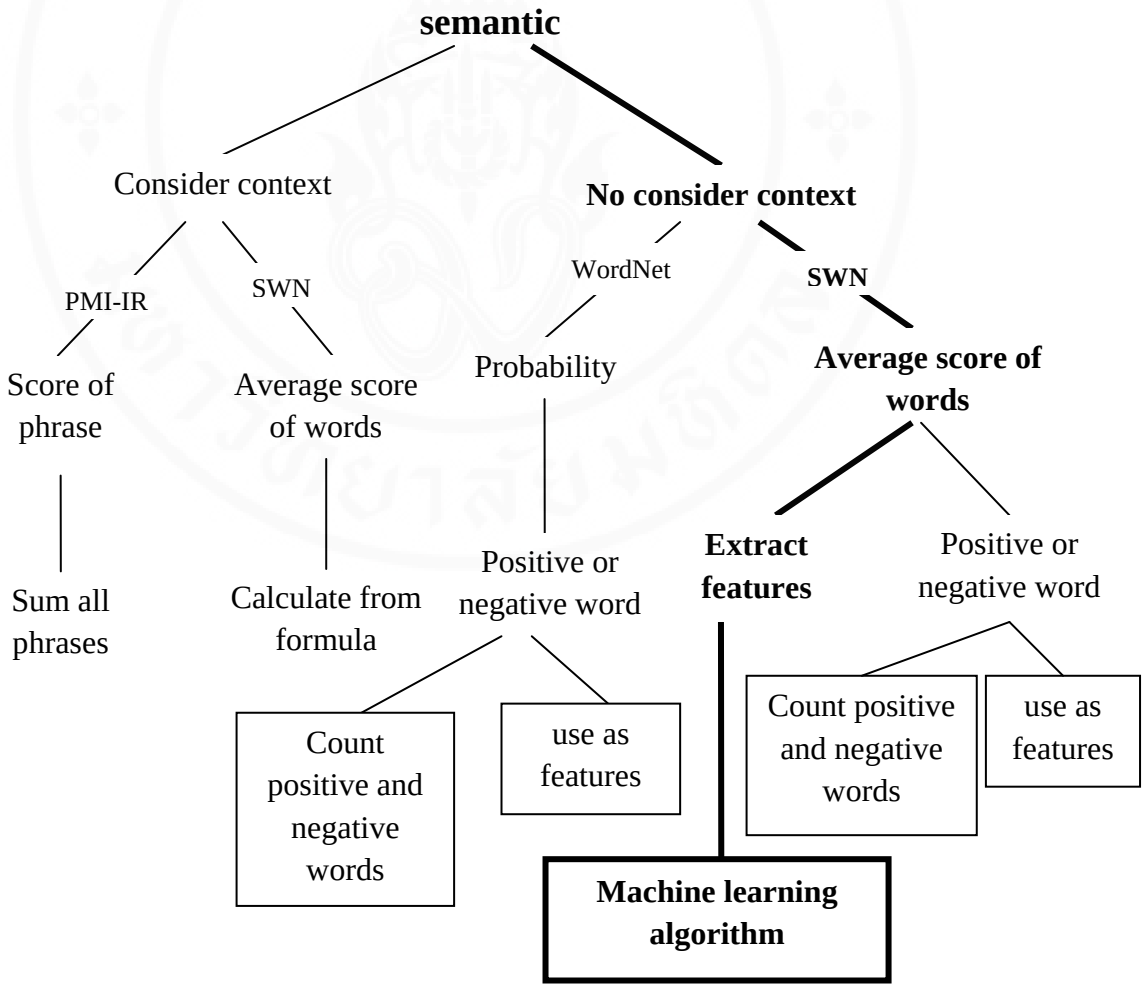


Figure 6.1 The method to obtain semantic orientation of words or phrases

Figure 6.1 described the details to obtain semantic orientation of words or phrases. The bold words and lines show the steps which obtains average scores of words for extracting features, and then using machine learning method to classify reviews in this study.

6.1 Concept of WorkConclusion

A word has three scores, i.e., positive, negative, and objective score. If the word is known as positive or negative word, the word can be chosen as one of training data. When a word classifier is constructed from performing on training data, unseen words with three scores can be predicted as positive or negative words.

In this study, classifying documents as positive or negative ones is concerned. Due to that, content words are components in documents and their functions are to express meaning, the parts of speech, i.e., verb, adjective, adverb, and noun are used in this study. Some words can be one more parts of speech which also give different senses. Therefore, the words must be categorized according to their POS and kept in the corresponding POS files of a document. Then, the words are assigned three scores from POS lists which are the same as POS of words. Scores of each POS are derived from words in that POS file. Each POS has three sentiment scores, i.e., positive, negative, and objective score, which each sentiment score is averaged from all words in that POS file. In order to have three types of sentiment and four concerned POS, thus there are 12 features, which are called "**sentiment features**." Initially, documents in training data are known as positive or negative ones. Furthermore, vectors of sentiment features are constructed as described above (in detail in chapter 4). Eventually, machine learning algorithms run through training data based on sentiment features and a sentiment classifier is constructed. Therefore, when unseen documents are transformed to vector of sentiment features and are executed on sentiment classifier, those documents are predicted as negative or positive ones. Moreover, the bag-of-words features are combined with sentiment features, thus the combination features come up. The combination features with learning algorithms are expected to improve accuracy from bag-of-words features with learning algorithms. Three machine learning algorithms are used, i.e., linear SVM, Naïve

Baye(multinomial), and Decision Tree (J48). Lexical analysis is also experimented to compare with learning algorithms on features. Furthermore, dimension reduction techniques, i.e., Relief Algorithm, PCA, and correlation are expected to select potential features and improve accuracy of classification.

6.2 Conclusion of experimental results

1.The combination features which are from combining new extracted features called "**sentiment features**" and bag-of-words features on linear SVM has the highest accuracy, **84.21%**, among all experiments.

2. Linear SVM on combination features have ability to improve accuracy from SVM on bag-of-words features, which is 83.21%, and further used as the sentiment classifier.

3. Relief Algorithm is able to rank features relevance to the target concept. When Relief Algorithm is used for ranking combination features, the top rank features which are potential for classifying reviews comes from sentiment features. Thus, sentiment features are potential when combined with bag-of-words features. The first rank is **negativity of adjective**. The second rank is **positivity of adjective**. The third rank is **positivity of adverb**.

4.When using statistical test to evaluate the confidential level which combination features has more accuracy than bag-of-words features, the result shows that SVM with combination features has higher accuracy than the one at the confidence level **83.28%** by one-tailed test hypothesis.

5.The result of Naïve Bayes (multinomial) on sentiment features, which the accuracy is **70.40%**, is quite good and potential because only 12 features are used and take short time in training sentiment classifier.

6.Effectiveness of both recall and precision is measured by F-measure. F-measure for negative and positive classification of SVM on combination features are the highest among the other ones, separately compared by negative and positive classification. Furthermore, **negative classification** is effective than positive classificationof SVM on combination features.

7. Effectiveness of both positive and negative classification by SVM on combination features improve from SVM on bag-of-words features at level of significant 0.1.

8. The accuracy on 278 unseen reviews of sentiment classifier based on SVM on combination features is **85.25%**. The accuracy is higher than SVM on combination features estimated by five-fold cross-validation.

9.Relief Algorithm is expected to select the potential features but features which are select at various proportions of all features do not improve accuracy of classification. This result may be that all features are important and useful for classifying reviews. In addition, PCA cannot run due to usage large memory and software in this study has limitation.

10. In this study, Dataset is used the same one as the work of Pang et al [16]. The best accuracy of the work of Pang et al is 82.9% on SVM with bag-of-words features. In this study, the best accuracy is 84.21% on SVM with combination features which sentiment features are combined with bag-of-words features.

6.3 Recommendations

Negation word, e.g., not, never, no, will be paid more attention. In further study, there may be one attribute for negation words which is a special attribute for collecting frequency of negation words of each instance. Due to reversal meaning of the sentence from negation words, negation word attribute is expected to reflect opposite meaning of the sentence. The attribute may improve accuracy of sentiment classification.

The phrases which often occur in document collection and word bigrams are used as features for machine learning algorithms. Context will keep meaning of sentences, thus the phrases and word bigrams can improve accuracy of sentiment classification.

REFERENCES

1. Kennedy A. and Inkpen D., Sentiment Classification of Movie and Product Reviews Using Contextual Valence Shifters., *Computational Intelligence*, 2006, Vol.22, 110-125,
2. Joachims T., Text Categorization with Support Vector Machines: Learning with Many Relevant Features., *Proceedings of ECML-98, 10th European Conference on Machine Learning*, 1998, pp.137-142
3. McCallum A. and Nigam K., A Comparison of Event Models for Naïve Bayes Text Classification., *Proceedings AAAI-98 Workshop on Learning for Text Categorization*, 1998, pp.41-48, AAAI Press
4. Apte C., Damerau F., and Weiss S.M., Text Mining with Decision Trees and Decision Rules., *Conference on Automated Learning and Discovery*, Carnegie-Mellon University, 1998
5. Pang B. and Lee L., Movie Review Data, *polarity dataset v1.1* [Online]. Available from: URL: http://www.cs.cornell.edu/people/pabo/movie-review-data/
6. Na J.-C., Khoo C., and Wu P.H.J., Use of the negation phrases in automatic sentiment classification of product reviews., Library Collections, Acquisitions, & Technical Services 29, 2005, pp.180-191, ELSEVIER
7. Liu B., Web Data Mining., 2007, Springer
8. Hu M. and Liu B., Mining and Summarizing Customer Reviews., *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004, pp.168-177
9. Turney P.D., Thumb Up or Thumb Down? Semantic Orientation Applied to Unsupervised Classification of Reviews., *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 2002, pp.417-424
10. Meena A. and Prabhakar T.V., Sentence Level Sentiment Analysis in the Presence of Conjuncts Using Linguistic Analysis., *Proceedings of the 29th European conference on IR research*, 2007, pp.573-580

11. Dang Y., Zhang Y., and Chen H., A Lexicon-Enhanced Method for Sentiment Classification: An Experiment on Online Product Reviews., *Journal IEEE Intelligent Systems*, Vol 25, 2010, pp.46-53
12. Salton G., Wong A., and Yang C.S., A vector space model for automatic indexing., *Communications of the ACM*, No.18, Vol.11, 1975, pp. 613-620
13. Sebastiani F., Machine Learning in Automated Text Categorization, *ACM Computer Surveys*, Vol.34, 2002, pp. 1-47
14. Scott S. and Matwin S., Feature Engineering for Text Classification., *Proceedings of the Sixteenth International Conference on Machine Learning*, 1999, pp. 379-388
15. Ohana B. and Tierney B., Sentiment Classification of Reviews Using Sentiwordnet.,9^{th}. *IT & T Conference*, Dublin Institute of Technology, Dublin, Ireland, 2009
16. Pang B., Lee L., and Vaithyanathan S., Thumb up? Sentiment Classification using Machine Learning Techniques., *Proceedings of conference on Empirical methods in natural language processing*, Vol 10, 2002, pp. 79-86
17. Matsumoto S., Takamura H., and Okumura M., Sentiment Classification Using Word Sub-Sequences and Dependency Sub-Trees., *Proceedings of the* 9^{th} *PAKDD*, 2005, pp.301-311
18. Finn A. and Kushmerick N., Learning to classify documents according to genre., *Journal of the American Society for Information Science and Technology*, Vol 57, 2006, pp.1506-1518
19. Tianxia G., Processing Sentiments and Opinions in Text: A Survey.,[Online]. From URL:www.comp.nus.edu.sg/~gongtian/CS6207/CS6207_Survey-.pdf, 2008
20. Hotho A., Nurnberger A., and Paaß G., A Brief Survey of Text Mining., LDV Forum - GLDV Journal for Computational Linguistics and Language Technology, 2005
21. Baeza-Yates R. and Ribeiro-Neto B., Modern Information Retrieval., 1999, the ACM press. ADDISON WESLEY
22. Stanford Tokenizer [Online] Available from URL: http://nlp.stanford.edu/software/tokenizer.shtml

23. Gibbon D.C. and Liu Z., Introduction to video search engines., *ACM computing classification*, 2008, Springer Berlin Heidelberg
24. Mitchell T.M., Machine Learning,1997, McGraw-Hill International
25. Fisher R.A., The use of multiple measurements in taxonomic problems., *Annual Eugenics, 7 (Part II)*, 1936, pp. 179-188
26. Harrison D. and RubinfeldD.L., Hedonic prices and the demand for clean air., *Journal of Environment, Economics and Management*, 1978, pp.81-102
27. Burbidge R. and Buxton B., An Introduction to Support Vector Machines for Data Mining., the INTErSECT Faraday Partnership and the National Physical Laboratory, 2001
28. Analytics: Decision tree [online] Available from URL: http://gautam.lis.illinois.edu/monkmiddleware/public/analytics/-decisiontree.html
29. Kivinen J., Warmuth M., and Auer P., The perceptron algorithm vs. winnow:Linear vs. logarithmic mistake bounds when few input variables are relevant., In*Conference on Computational Learning Theory*, 1995
30. Berwick R., An Idiot's guide to Support vectormachines (SVMs) [online] Available from URL:http://www.cs.ucf.edu/courses/cap6412/fall2009/-papers/Berwick2003.pdf
31. Kira K. and Rendell L.A., A practical approach to feature selection., *Proceedings of the ninth international workshop on Machine learning*, 1992, pp. 249-256
32. JolliffeI.T., Principal Component Analysis, second edition., 2002, Springer
33. Smith L.I.,A tutorial on Principal Components Analysis., 2002, [online] Available from URL: http://www.sccg.sk/~haladova/principal_components.pdf
34. Esuli A., Sebastiani F., 2005, Determining the semantic orientation of terms through gloss analysis., In proceedings of CIKM-05, 14th ACM International Conference on Information and Knowledge Management, pages 617-624, Bremen, DE.
35. Esuli A., Sebastiani F., 2006, SentiWordNet: A publicly available lexical resource for opinion mining., In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC'06), pages 417-422, Genova, IT

36. Miller G.A., WordNet: A Lexical Database for English., Communications of the ACM,1995
37. Sebastiani F., SentiWordNet 3.0., [online] Available from URL: http://sentiwordnet.isti.cnr.it/
38. Witten I.H., Frank E., Data Mining Practical Machine Learning Tools and Techniques.,2005, ELSEVIER.
39. Croft W.B., Metzler D., and Strohman T., Search Engines Information Retrieval in Practice, 2010, Pearson International Edition.
40. Kasemsritanawat T., Phienthrakul T., Sentiment Classification using Bayes' Classifier and Feature Selection with Relief Algorithm., National Conference on Computer Information Technologies (CIT 2011), NakornPathom, Thailand.
41. Santorini B., Part-of-Speech Tagging Guidelines for the Penn Treebank Project (3rd Revision)., [online] from URL : http://repository.upenn.edu/cis_reports/ 570, 1990.

APPENDICES

APPENDIX A

PENN TREEBANK POS TAGSET

1. CC	Coordinating conjunction	25. TO	*to*
2. CD	Cardinal number	26. UH	Interjection
3. DT	Determiner	27. VB	Verb, base form
4. EX	Existential *there*	28. VBD	Verb, past tense
5. FW	Foreign word	29. VBG	Verb, gerund/present participle
6. IN	Preposition/subordinating conjunction	30. VBN	Verb, past participle
7. JJ	Adjective	31. VBP	Verb, non-3rd ps. sing. present
8. JJR	Adjective, comparative	32. VBZ	Verb, 3rd ps. sing. present
9. JJS	Adjective, superlative	33. WDT	*wh*-determiner
10. LS	List item marker	34. WP	*wh*-pronoun
11. MD	Modal	35. WP$	Possessive *wh*-pronoun
12. NN	Noun, singular or mass	36. WRB	*wh*-adverb
13. NNS	Noun, plural	37. #	Pound sign
14. NNP	Proper noun, singular	38. $	Dollar sign
15. NNPS	Proper noun, plural	39. .	Sentence-final punctuation
16. PDT	Predeterminer	40. ,	Comma
17. POS	Possessive ending	41. :	Colon, semi-colon
18. PRP	Personal pronoun	42. (	Left bracket character
19. PP$	Possessive pronoun	43.)	Right bracket character
20. RB	Adverb	44. "	Straight double quote
21. RBR	Adverb, comparative	45. '	Left open single quote
22. RBS	Adverb, superlative	46. "	Left open double quote
23. RP	Particle	47. '	Right close single quote
24. SYM	Symbol (mathematical or scientific)	48. "	Right close double quote

PENN TREEBANK POS TAGSET [40]

APPENDIX B

Review Classification using Part-Of-Speech Sentiment Strength Learning

Titima Kasemsritanawat[1] and Tanasanee Phienthrakul[2]

[1]Technology of Information System Management,
Faculty of Engineering, Mahidol University
[2]Computer Engineering, Faculty of Engineering,
Mahidol University
[1]mom_love_za@hotmail.com, [2]tanasanee.phi@mahidol.ac.th

Abstract

This paper proposes the new extracted features for sentiment classification. These new features are derived from SentiWordNet and they are called "part-of-speech (POS) sentiment strength features". These features are described by the strength of positivity, negativity, and objectivity for each POS. Support vector machines (SVM) and multinomial Naïve Bayes are used to classify the reviews on the extracted features. The set of proposed features and bag-of-words features are evaluated on movie reviews. The experiment results showed that using the combination of bag-of-words and POS sentiment strength features gives the highest accuracy when SVM were applied. Furthermore, the accuracies of both SVM and multinomial Naïve Bayes on POS sentiment strength features are higher than the accuracy of the lexicon analysis in review classification.

Keyword: Sentiment Classification, Part-Of-Speech, Movie Reviews, Feature Extraction

1. Introduction

There is a lot of information on the Internet. Users can search on the Internet for knowledge, information, or entertainment. Many websites allow users to review products, services, or movies. These reviews are useful for the readers who want to buy or choose one. However, if there are too many reviews and each review is long, the readers will take a lot of time to read all reviews. Also, the readers may bias to some reviews which make them miss in decision.

Due to such problems, the researches on opinion summarization and review classification are coming up topics. Many opinions were classified into "recommended" or "not recommended". The specific names of these researches can be called "opinion mining", "sentiment analysis", or "sentiment classification". There are several techniques that were proposed for sentiment classification. In the research of Hu et al. [1], the product features were extracted and the user opinions were identified to be positive or negative based on the product features. The sentences with one or more features and one or more opinion words are considered to be the opinion sentences. Only positive or negative orientations of adjectives or opinion words were used in their research.

In the work of Turney [2], point-wise mutual information (PMI) and information retrieval (IR) algorithms were used to estimate semantic orientation of each extracted phrases. The similarities of words or phrases are measured. Class of each review was assigned by the numerical rating. The rating indicated the intensity or strength of the semantic orientation. Turney's work achieved an average accuracy of 66% for movie reviews.

The work of Kim et al. [3] developed a sentence-based sentiment classifier by considering the sentiment strength or orientation of words containing in the sentence. Their study identified the people who hold each sentiment which is called "holder" and determined opinion region within a sentence. Sentiment words within the holder-based regions of opinion are combined to produce holder's sentiment. In their study, lexicon of sentiment words was constructed, while synonym and anonym sets from WordNet were used to obtain the expansions of seed words as the work of Hu et al. [1].

The strength measures of sentiment words were developed and they were used in many researches for considering the absolutely positive or negative orientation. However, we notice that the accuracies of sentiment classification may be improved by cooperating machine learning techniques and sentiment strength measures. Instead of identifying whether the words are positive or negative, sentiment scores of these words will be used as the

features of the learning algorithm. Sentiment classification in this paper is performed at document level. The documents will be classified to be positive or negative opinions which are referred to recommend or not recommend respectively.

In this paper, the idea of sentiment classification, SentiWordNet, and machine learning techniques are briefly reviewed in section 2. Review classification by part-of-speech sentiment strength learning will be described in section 3. The details of the experiment and results in this study are illustrated in section 4, and all results will be concluded in section 5.

2. Background

In this section, the sentiment classification is described. Then, SentiWordNet that is a method for lexical analysis is explained. After that, the machine learning techniques for sentiment classification are briefly reviewed.

2.1 Sentiment Classification

Sentiment classification can be mainly divided into semantic approach and machine learning approach. The work of semantic approaches is initially based on sentiment words or phrases identification and also their sentiment or semantic orientation identification which is important for later sentence level or document level sentiment classification. The semantic orientation of a word indicates the direction that the word deviates from the norm from its group. Words that encode a desirable state have a positive orientation (e.g., beautiful and awesome), while words that represent undesirable state have a negative orientation (e.g., bad and disappointing) [1].

2.2 SentiWordNet

SentiWordNet is a lexical resource, which used WordNet [4] synonym set *s*. The synonym set is associated to three numeric scores, i.e., objective score, positive score, and negative score. These scores describe the characteristics of objective, positive, and negative of each term in the synonym set. The method used to develop SentiWordNet is based on the quantitative analysis of the glosses associated to synonym sets [5]. Each word may have several meaning or several senses. Positivity, negativity, and objectivity scores for a word from SentiWordNet can be calculated from the average scores of all senses of a word [6]. These scores can be calculated by equation (1)-(3).

$$score_{pos}(A,o) = \frac{1}{n}\sum_{i=1}^{n} score_{pos,o}(i), \quad (1)$$

$$score_{neg}(A,o) = \frac{1}{n}\sum_{i=1}^{n} score_{neg,o}(i), \quad (2)$$

$$score_{obj}(A,o) = \frac{1}{n}\sum_{i=1}^{n} score_{obj,o}(i), \quad (3)$$

where A is a word with n corresponding synonym sets on a part of speech, and $score_{\{pos|neg|obj\},o}(i)$ is the score of synonym sets i for term A at the part of speech o.

2.3 Machine Learning Techniques

There are many machine learning techniques that can be applied for sentiment classification. Support vector machine (SVM) is a technique that is better than other machine-learning algorithms [7]. Multinomial Naïve Bayes has also been used for text classification by numerous researches [8]. Thus, these two machine learning algorithms are interesting.

Pang et al. [9] examined the effectiveness of three machine learning algorithms, i.e., Naïve Bayes, Maximum Entropy, and Support Vector Machines on bag-of-words features. Word unigrams and word bigrams are used as bag-of-words features in their researches. The best performance is obtained by using SVM on unigrams bag-of-words.

3. POS Sentiment Strength Learning

This paper proposes the new features for sentiment classification. These new features are derived from SentiWordNet. The process of sentiment classification and POS sentiment strength feature extraction are described in the following.

3.1 Bag-Of-Word Features

A text document can be described by the set of words, which are obtained by using text pre-processing, i.e., sentence splitting, tokenization, elimination of stop word, stemming, and getting the distinct words. Bag-of-words features are also used in sentiment classification commonly, such as the work of Pang et al [9]. All words in documents are collected, and these words are used for sentiment classification. These features are called bag-of-words features. For each document, the frequencies of each word are found and they are used as the value of an input vectors. After sentence splitting and word tokenization, 309 stop words are ignored. Then, Porter's algorithm is applied to perform stemming.

3.2 POS Sentiment Strength Features

In sentiment classification, content words, i.e., nouns, verbs, adjectives, and adverbs, are used as sentiment words because these words can express positive, negative, or objective meanings [10]. In this paper, Stanford Log-linear Part-Of-Speech

Tagger[1] is used to obtain part-of-speech (POS) of each word.

The types of sentiment of a word are positivity and negativity, including objectivity which has no sentiment. In this paper, firstly, sentiment strength of all content words in SentiWordNet are assumed to be features which the word is described by its three values of sentiment strength, i.e., strength of positivity, negativity, and objectivity. However, due to the large amount of content words in the lexicon, the POS tagging is used for categorizing words according to their parts of speech. Instead of using strength of each sentiment type of a word as one feature, this paper uses strength of each sentiment type of each POS as one feature. Also, the four parts of speech which are already mentioned are considered to use due to their potential for sentiment-classification task.

Furthermore, if words are used in different parts of speech, so the sentiments are also different; For example, "love" in "I love this movie" gives high positive sentiment, while "love" in "This is a love story." gives no sentiment [9]. In the study of Finn et al. [11], the document is represented as vector of POS features due to their assumption that the POS statistics would reflect the style of the language sufficiently for the learning algorithm to distinguish between different genre classes. Each POS tag feature is described by its occurrence which is the percentage of the total number of words in the document. Due to give correct sense of the word in different parts of speech and imply the style of language in the document, POS feature is potential and useful for linguistic point of view. POS tagging is used for preparing POS feature. However, this study performs sentiment classification, thus sentiment strength (or intensity) per a POS is concerned rather than percentage of each POS.

Each word can have several meanings in the same POS depending on the contexts around the word, thus the average strength of each sentiment type of all meanings of the word with the same POS are used. The weights of positive, negative, and objective sentiments for each word in this paper are derived from SentiWordNet. The method for deriving average strength of each sentiment type of words is illustrated in section 2.2 and the related equations, i.e., equation (1),(2), and (3), are also in that section.

If a word with the specific POS in the document is found in the content word list, its three values of sentiment strength from content word list are assigned to the corresponding POS. For example, the word "enjoyable" which is an adjective in the document is found in the content word list, thus the word is obtained its sentiment strength, strength of positive sentiment as 0.5, strength of negative sentiment as 0.25, and strength of objectivity as 0.25. This process performs iteratively until all words with concerned POS in a document are inspected. The values of strength of each sentiment type of all words with the same POS are summed up and averaged which is referred as POS sentiment strength.

For the reasons above, strength of each sentiment type of each POS is used as one feature which is described by an average strength of the sentiment of the words which have the corresponding POS in a document as shown in Table1. In order to have three types of sentiment and four concerned POS, thus there are 12 features which are called as "part-of-speech (POS) sentiment strength features". Due to perform this technique, the training dataset are called "Part-of-speech sentiment strength vectors".

Table 1. An Example of POS Sentiment Strength Features and their Value

Sentiment Strength / Part of Speech	Positivity	Negativity	Objectivity
VERB	0.5	0.2	0.3
ADJECTIVE	0.7	0.2	0.1
ADVERB	0.4	0.3	0.3
NOUN	0.2	0.1	0.7

3.3 Review Classification

POS sentiment strength features and bag-of-words features are extracted from movie reviews. These features are used as input for machine learning algorithms. In this paper, linear SVM and multinomial Naive Bayes are considered to classify the reviews into positive or negative classes. Moreover, these learning techniques are compared to the lexicon analysis.

4. Experimental Results

In order to evaluate the proposed features, the features of movie reviews are extracted from POS sentiment strength. The experimentation and results are reported in this section.

4.1 Experimentation

In this study, movie reviews from Cornell website[2] are brought as the dataset for the algorithms on each feature set to classify reviews as recommended or not recommended. There are 1108

[1] http://nlp.stanford.edu/software/tagger.shtml

[2] http://www.cs.cornell.edu/people/pabo/movie-review-data/, alternative version by Nathan Treloar

reviews, 554 positive reviews and 554 negative reviews, for investigating the effectiveness of using machine learning algorithms with sets of features.

In this paper, there are 19,115 words for bag-of-words features. The word frequency is used to describe each word feature. Due to investigate the effectiveness of classifying documents by using POS sentiment strength features on machine learning algorithms, the lexicon analysis is used to compare with these features which are run on machine learning algorithms. The method of lexical analysis is as followings :

In a document, the negativity and positivity of words which are derived from SentiWordNet are used to determine orientation of words by calculating the difference between positivity and negativity of some words in a document. The words used in this lexical analysis are the same ones used in extracting POS sentiment strength features which are categorized according to their parts of speech and obtained the negativity, positivity scores which are also relative to corresponding POS from SentiWordNet.

For considering the word orientation, if the strength of the word is below zero, the word has negative orientation. If the strength of the word is above zero, the word has positive orientation. If the strength of the word is zero, the word has neutral orientation. The strength of all concerned words are summed up and averaged. To classify a document, if the sign of average strength is minus, the document is assigned into negative class, and if the sign of average strength is plus, the document is assigned into positive class. In the case of the average strength of the document is zero, the result is neutral which is considered as wrong answer because the training dataset are labelled their classes only as positive or negative, not neutral.

In order to determine the potential of sentiment strength, machine learning algorithms run on the POS sentiment strength features. In this paper, sentiment strength is expected to improve accuracy of sentiment classification from using only bag-of-words features which are common representation of a document. Thus, the combination of POS sentiment strength with bag-of-words features is compared its accuracy with only the bag-of-words features, based on machine learning algorithms

4.2 Results

In order to evaluate the accuracy in classifying documents into positive or negative classes of machine learning algorithms running on training data, five-fold cross-validation is used. The accuracies are illustrated in Table 2. The accuracy of using the machine learning algorithms running through POS sentiment strength features is 69.68% on SVM and 70.40% on multinomial Naïve Bayes. The lexical analysis has accuracy 65.79%. Thus, using machine learning with POS sentiment strength features has higher accuracy than the lexical analysis.

Bag-of-words features are used in machine learning algorithms. The accuracy in classifying documents into positive or negative classes is good, 83.12% on SVM and 79.69% on multinomial Naïve Bayes. Moreover, POS sentiment strength features are used to combine with bag-of-words features to improve the accuracy of the classification. The result is slightly increased from using only bag-of-words features, 84.21% on SVM and 79.87% on multinomial Naïve Bayes.

Table 2. Accuracy of Sentiment Classification

Features \ Machine Learning	SVM	Multinomial Naïve Bayes
Lexicon Analysis	65.79%	
POS Sentiment Strength Features	69.68%	70.40%
Bag-of-Words Features	83.12%	79.69%
Combination of Bag-of-Words and POS Sentiment Strength Features	**84.21%**	**79.87%**

5. Conclusion

The best accuracy is 84.21% from using SVM running on combination of bag-of-words and POS sentiment strength features. This result implies a good trend when adding POS sentiment strength features to set of bag-of-words features. POS sentiment strength features for multinomial Naive Bayes has slightly higher accuracy than SVM in classifying documents into positive or negative classes. Moreover, running machine learning algorithms on POS sentiment strength features has higher accuracy than the lexical analysis in classifying documents into positive or negative classes. These mean that both machine learning algorithms and POS sentiment strength features can improve the efficiency of sentiment classification.

References

[1] M. Hu and B. Liu, "Mining and Summarizing Customer Reviews," *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004, pp.168-177.

[2] P.D. Turney, "Thumb Up? or Thumb Down? Semantic Orientation Applied to Unsupervised Classification of Reviews," *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 2002, pp.417-424.

[3] S.-M. Kim and E. Hovy, "Determining the Sentiment Opinions," *Proceedings of the COLING conference*, Geneva, 2004.

[4] G.A. Miller, "WordNet: A Lexical Database for English," ACM, 1995, pp. 39-41.

[5] A. Esuli and F. Sebastiani, "SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining," *In Proceeding of the 5th Conference on Language Resources and Evaluation (LREC'06)*, Genova, Italy, 2006, pp. 417-422.

[6] K. Denecke, "Using SentiWordNet for Multilingual Sentiment Analysis," *In Proceeding of IEEE 24th International Conference on Data Engineering Workshop*, Hanouver, Germany, April 2008, pp. 507-512.

[7] J.-C Na, C Khoo, and P.H.J Wu, "Use of negation phrases in automatic sentiment classification of product reviews," Library Collections, Acquisitions, & Technical Services 29, ELSEVIER, 2005, pp.180-191.

[8] A. McCallum and K. Nigam, "A Comparison of Event Models for Naïve Bayes Text Classification," *In AAAI-98 Workshop on Learning for Text Categorization*, 1998.

[9] B. Pang, L. Lee, and S. Vaithyanathan, "Thumb up? Sentiment Classification using Machine Learning Techniques." *Proceedings of conference on Empirical methods in natural language processing*, 2002, pp. 79-86.

[10] B. Liu, Web data mining, Springer, 2007.

[11] A. Finn and N. Kushmerick, "Learning to classify documents according to genre," Journal of the American Society for Information Science and Technology, 2006, pp. 1506-1518.

Some parts of this study are accepted to be oral presentation at International Computer Science and Engineering Conference (ICSEC 2012) Pattaya, Thailand on 17-19 October 2012.

การจำแนกความคิดเห็นโดยใช้ตัวจำแนกแบบเบย์ร่วมกับการเลือกคุณลักษณะด้วยอัลกอริทึมรีลีฟ

Sentiment Classification using Bayes' Classifier and Feature Selection with Relief Algorithm

ฐิติมา เกษมศรีธนาวัฒน์[1] และ ธนัสนี เพียรดระกูล[2]

[1]สาขาวิชาเทคโนโลยีการจัดการระบบสารสนเทศ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยมหิดล

[2]ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยมหิดล

[1]mom_love_za@hotmail.com และ [2]tanasanee@yahoo.com

Abstract

This paper purposes the method to summarize the book reviews as "thumbs up" or "thumbs down". Machine learning, i.e. Naïve Bayes, Decision Tree, and Multi-Layer Perceptron, are used to classify the reviews. These learning algorithms are evaluated on book reviews. The experimental results showed that Naïve Bayes gives higher performance than the other algorithms. Furthermore, feature selection algorithms, i.e. principle components analysis (PCA) and Relief algorithm, are used in preprocessing step in order to increase the performance of learning methods. We also found that Relief algorithm can improve the performance of sentiment classification.

Keywords: Sentiment Classification, Naïve Bayes, Relief Algorithm, Product Reviews

บทคัดย่อ

บทความนี้นำเสนอวิธีการสรุปความเห็นจากทัศนคติที่เกี่ยวข้องกับหนังสือว่ามีทัศนคติที่ดี หรือไม่ดีต่อหนังสือนั้นๆ โดยใช้การเรียนรู้ของเครื่อง อันได้แก่ Naive Bayes, Decision Tree, Multi-Layer Perceptron เพื่อจำแนกความเห็น อัลกอริทึมเหล่านี้ถูกประเมินความแม่นยำของการเรียนรู้บนทัศนคติที่เกี่ยวข้องกับหนังสือ ผลการทดสอบพบว่า Naive Bayesให้ความแม่นยำที่ดีกว่าวิธีอื่น นอกจากนั้นในงานนี้ได้คัดเลือกคุณลักษณะโดยใช้ Principle Components Analysis และ Relief algorithm เพื่อเพิ่มประสิทธิภาพในช่วงการเรียนรู้ด้วย ซึ่งพบว่า Relief algorithm สามารถเพิ่มประสิทธิภาพในการจำแนกความคิดเห็นได้

คำสำคัญ การจำแนกตามความรู้สึก, เบย์อย่างง่าย, รีลีฟอัลกอริทึม, ความคิดเห็นต่อสินค้า

1.บทนำ

เนื่องจากปัจจุบันนี้เป็นยุคที่มีข้อมูลข่าวสารบนอินเทอร์เน็ตเป็นจำนวนมาก ทำให้ผู้ใช้งานอินเทอร์เน็ตต้องใช้เวลามากในการเลือกข้อมูลที่ตรงกับความต้องการของตนเองและใช้เวลาอ่านข้อมูลนาน จึงทำให้เกิดงานวิจัยสาขาต่างๆเพิ่มขึ้นมา อาทิ เช่น การสรุปใจความสำคัญของเอกสารอัตโนมัติ

(Automatic Text Summarization) เป็นต้น และมาในระยะหลังนี้มีหลายเว็บไซต์ได้ให้ผู้ใช้สินค้าต่างๆได้มาวิจารณ์สินค้าผ่านทางเว็บไซต์ ทำให้เป็นประโยชน์ต่อผู้ที่เข้ามาชมเว็บไซต์เพื่อประกอบการตัดสินใจเลือกซื้อสินค้าและบริการ แต่การวิจารณ์หรือให้ความเห็นดังกล่าวมีเป็นจำนวนมากทำให้ผู้เข้าชมต้องใช้เวลาอ่านมากซึ่งอาจทำให้ลดความสนใจในการอ่านข้อความวิจารณ์ ดังนั้นการสรุปทัศนคติที่มีต่อสินค้านั้นจะช่วยขจัดปัญหาการใช้เวลาในการอ่านลงได้บ้างเพื่อประกอบการพิจารณาเลือกสินค้า

ซึ่งกลไกการทำงานต้องมีการจำแนกความเห็นตามทัศนคติก่อนว่าเป็นทัศนคติที่ดีหรือไม่ดีต่อสินค้าหรือบริการนั้นๆซึ่งจัดเป็นงานด้านการจำแนกตามความรู้สึก (Sentiment Classification) ซึ่งนอกจากช่วยด้านประหยัดเวลาแล้วยังสามารถเพิ่มความดึงดูดด้วยภาพลักษณ์ที่ทันสมัยดังเช่นเว็บไซต์ www.rottentomatoes.com

ในบทความนี้ผู้เขียนสนใจวิธีจำแนกเอกสารตามทัศนคติที่มีต่อสินค้าและบริการแบบอัตโนมัติโดยใช้ Machine Learning Techniques ซึ่งวิธีการเรียนรู้ที่นำมาใช้ได้แก่ Naïve Bayes, Decision Tree (J48), Multi-Layer Perceptron โดยที่ข้อมูลที่นำมาทำการทดลองมาจากความคิดเห็นเกี่ยวกับหนังสือคอมพิวเตอร์ด้านโปรแกรมมิ่งของเว็บไซต์ http://www.amazon.com

นอกจากนี้แล้วผู้เขียนยังได้เพิ่มขั้นตอนการคัดเลือกคุณสมบัติของข้อมูล (Feature Selection Algorithm) ในขั้นตอนการเตรียมข้อมูล (Preprocessing Step) ด้วยเพื่อเพิ่มความแม่นยำต่อการฝึกการเรียนรู้ให้แก่ข้อมูลซึ่งขั้นตอนในการทำงานจะกล่าวในหัวข้อที่ 3

2.วิจารณ์วรรณกรรม

งานวิจัยทาง Sentiment Classification ในปัจจุบันแบ่งออกเป็น 2 เทคนิคใหญ่ๆ [1] ได้แก่ การเรียนรู้โดยพิจารณาจากความหมายของคำ (Semantic Orientation) และเทคนิคการเรียนรู้ด้วยเครื่อง (Machine Learning Techniques) ดังตัวอย่างของงานต่อไปนี้

การเรียนรู้โดยพิจารณาความหมายของคำมีปรากฏในงานของ Turney,2002[2] ซึ่งใช้ความเห็นมาจากเว็บไซต์ www.epinions.com ซึ่งได้นำหมวดโทรศัพท์มือถือ, ธนาคาร, ภาพยนตร์ และการท่องเที่ยวมาจำแนกทัศนคติของผู้เขียนว่าดีหรือไม่ดี โดยคัดเลือกเฉพาะวลีที่มีคำคุณศัพท์ หรือคำวิเศษณ์จากความเห็นแล้วใช้อัลกอริทึม Point-wise Mutual Information และ Information Retrieval (PMI-IR) ซึ่งทำการหาค่าความหมายของวลี (ดีหรือไม่)โดยเปรียบเทียบกับคำศัพท์ที่ช่วยในการจำแนก (Excellent และ Poor) จากนั้นจึงหาค่าเฉลี่ยทัศนคติของวลีในความเห็นนั้นๆ ซึ่งจะได้ผลว่าความเห็นนั้นเป็นทัศนคติที่ดี หรือไม่ดีซึ่งการจำแนกให้ผลถูกต้องเป็นที่น่าพอใจ แต่ในงานนี้มีข้อจำกัดด้านการใช้เวลานานในช่วงการรอผลจาก Search Engine ในการเปรียบเทียบความหมายของวลีกับคำศัพท์ รวมถึงไม่ได้ใช้ปัจจัยอื่นๆที่มีผลต่อการจำแนกทัศนคติ นอกเหนือจากการพิจารณาความหมายของคำ

ตัวอย่างงานวิจัยที่ใช้การเรียนรู้ด้วยเครื่อง เช่นงานของ Pang,2002[3] ซึ่งใช้ความเห็นต่อภาพยนตร์มา 1400 ความเห็นจากเว็บไซต์ http://reviews.imdb.com/Reviews/ และเปลี่ยนค่าเรตติ้งของผู้ให้ความเห็นมาเป็นทัศนคติที่ดี ไม่ดี หรือเป็นกลางต่อภาพยนตร์นั้นๆ จากนั้นจะเตรียมคุณลักษณะ ซึ่งจะใช้ Unigram, Bigram นำมาเปรียบเทียบกันว่าใช้คุณลักษณะแบบใดให้ค่าความถูกต้องในการจำแนกความเห็นตามทัศนคติได้ดีที่สุด ส่วนขั้นตอนการสอน ใช้ learning method 3 ชนิด ได้แก่ Naïve Bayes, Maximum Entropy และ Support Vector Machines มาเปรียบเทียบกัน ซึ่งพบว่าเมื่อใช้คุณลักษณะเป็น Unigram โดยพิจารณาเฉพาะพบ หรือไม่พบคุณลักษณะ (ไม่คำนึงถึงความถี่ของการพบคุณลักษณะ) และใช้ Learning Method เป็น Support Vector Machines จะให้

ความถูกต้องในการจำแนกความเห็นตามทัศนคติสูงสุด สำหรับในงานวิจัยนี้สนใจเทคนิคการเรียนรู้ด้วยเครื่อง โดยพิจารณาใช้เทคนิคการลดข้อมูลแบบ Relief ร่วมกับ Bayes' Classifier

2.1 Relief Algorithm

Relief เป็นอัลกอริทึมที่ได้รับความสนใจต่อเนื่องมาจากการเรียนรู้ของ Instance-based Learning ซึ่งได้ใช้การถ่วงน้ำหนักของคุณลักษณะ Relief ใช้การหาคุณลักษณะที่มีความเกี่ยวข้องกับ Target Concept โดยอาศัยหลักการทางสถิติและมีประสิทธิภาพของการทำงานที่ดี การทำงานของ Relief ใช้เวลาเป็นเชิงเส้นตรง (Linear Time)ซึ่งขึ้นกับจำนวนของคุณลักษณะ และจำนวนตัวอย่างของข้อมูลที่ใช้ในการเรียนรู้ จำนวนของคุณลักษณะที่ผ่าน Relief มักน้อยลงอันเนื่องมาจากอัลกอริทึมได้คัดเลือกคุณลักษณะที่มีความเกี่ยวข้องกับ Target Concept (Relevant Feature) และละทิ้งคุณลักษณะที่ไม่มีความเกี่ยวข้องกับ Target Concept [4]

จุดเด่นของ Relief คือทนทานต่อสิ่งรบกวน (Noise-Tolerant) และไม่ถูกกระทบจากการที่คุณลักษณะมีความสัมพันธ์ระหว่างกัน (Feature Interaction) เพราะโดย ปกติแล้วคุณลักษณะมักมีความสัมพันธ์ระหว่างกัน อัลกอริทึมของ Relief แสดงดังรูปที่ 1 เมื่อ S คือข้อมูลที่ใช้ในการเรียนรู้, m เป็นจำนวนรอบของการสุ่มตัวอย่าง, τ คือ Threshold ของความเกี่ยวข้องกับ Target Concept และ p คือจำนวนคุณลักษณะ Relief ใช้ระยะทางของยูคลิด p มิติ ในการเลือก Near-Hit และ Near-Miss ในแต่ละรอบน้ำหนักของคุณลักษณะ W จะถูกอัพเดท และสุดท้ายจะพิจารณาค่าเฉลี่ยน้ำหนักของแต่ละคุณลักษณะซึ่งจะดูว่าถึงค่าThreshold ที่กำหนดหรือไม่ ถ้าถึงระดับ Threshold จะเป็นคุณลักษณะที่มีความเกี่ยวข้องกับ Target Concept แต่ถ้าไม่ถึงระดับ Threshold จะเป็นคุณลักษณะที่ไม่มีความเกี่ยวข้องกับ Target Concept

When x_k and y_k are nominal,

$$diff(x_k, y_k) = \begin{cases} 0 & \text{<if } x_k \text{ and } y_k \text{ are the same>} \\ 1 & \text{<if } x_k \text{ and } y_k \text{ are different>} \end{cases}$$

When x_k and y_k are numerical,

$diff(x_k, y_k)=(x_k - y_k)/nu_k$

where nu_k is a normalization unit to normalize the values of diff into the interval [0, 1]

```
Relief(S, m, τ)
    Separate S into S+ = {positive instances} and
        S- = {negative instances}
    W = (0, 0, . . . , 0)
    For i = 1 to m
        Pick at random an instance X ∈ S
        Pick at random one of the positive instances
            closest to X, Z+ ∈ S+
        Pick at random one of the negative instances
            closest to X, Z- ∈ S-
        if (X is a positive instance)
            then  Near-hit = Z+; Near-miss = Z-
            else  Near-hit = Z-; Near-miss = Z+
        update-weight(W, X, Near-hit, Near-miss)
    Relevance = (1/m)W
    For i = 1 to p
        if (relevance_i ≥ τ)
            then  f_i is a relevant feature
            else  f_i is an irrelevant feature

update-weight(W, X, Near-hit, Near-miss)
    For i = 1 to p
        W_i = W_i - diff(x_i, near-hit_i)^2 + diff(x_i, near-miss_i)^2
```

รูปที่ 1: Relief algorithm [4]

2.2 Naïve Bayes' Classifier

Naïve Bayes เป็นอัลกอริทึมที่ถูกนำมาใช้อย่างแพร่หลายในงานจำแนกเอกสาร และให้ผลที่ดี การจำแนกโดย Naïve Bayes เริ่มจากแต่ละอินสแตนซ์ x ซึ่งจัดอยู่ในรูปเวกเตอร์ของค่าคุณลักษณะทุกคุณลักษณะ ดังนี้ $<a_1, a_2 \ldots a_n>$ โดยที่ Target Value ของแต่ละอินสแตนซ์เป็นค่าใดๆ ภายใน เซต V (V มีสมาชิกเป็นค่า Target Value ต่างๆ) [5]

การเรียนรู้โดยใช้การจำแนกโดย Naïve Bayes สุดท้ายแล้วจะได้ความน่าจะเป็นเงื่อนไขของแต่ละคุณลักษณะที่ใช้ในการทำนาย target value หรือ class ของอินสแตนซ์ใหม่ที่เข้ามาใช้การจำแนกชนิดนี้ อนึ่งการจำแนกโดย Naïve Bayes จะเป็นไปตามสมการดังนี้ [5]

$$V_{MAP} = \underset{v_j \in V}{\operatorname{argmax}} P(v_j | a_1, a_2 \ldots . a_n) \quad (1)$$

$$V_{MAP} = \underset{v_j \in V}{\operatorname{argmax}} \frac{P(a_1, a_2 \ldots . a_n | v_j) P(v_j)}{P(a_1, a_2 \ldots , a_n)} \quad (2)$$

$$= \operatorname{argmax}_{v_j \in V} P(a_1, a_2 \ldots . a_n | v_j) P(v_j)$$

ในการพิจารณาสมการดังกล่าวให้มาใช้งานง่ายขึ้น จะใช้สมมติฐานที่ แต่ละคุณลักษณะที่ทราบ Target Value $P(a_i|v_j)$ จะเป็นอิสระต่อกัน จะได้สมการจากสมมติฐานนี้เป็นดังนี้ Naïve Bayes Classifier:

$$V_{MAP} = \underset{v_j \in V}{\operatorname{argmax}} P(v_j) \prod_i P(a_i | v_j) \quad (3)$$

3.การจำแนกความเห็น

ในงานนี้ได้รวบรวมความเห็นของผู้ใช้หนังสือคอมพิวเตอร์ซึ่งเขียนเป็นภาษาอังกฤษจากเว็บไซต์ http://www.amazon.com โดยเลือกจากหมวดการเขียนโปรแกรมจาวา รวมทั้งหมด 100 ความเห็น ซึ่งประกอบด้วยทัศนคติที่ดี 57 ความเห็น และทัศนคติที่ไม่ดี 43 ความคิดเห็น (การให้ทัศนคติต่อสินค้าและบริการดังกล่าวมาจากการพิจารณาจากผู้เขียนงานนี้) นำความเห็นที่ได้ทั้งหมดมาจัดทำรายการคำศัพท์และเครื่องหมาย จากนั้นจะนำ +, -, 's, 've, 're, article (a, an, the) และตัวเลขโดดๆ ออกจากรายการคำศัพท์และเครื่องหมาย โดยที่เครื่องหมาย เช่น ?, ! ไม่นำออกจากรายการคำศัพท์และเครื่องหมาย เนื่องจากในงานนี้เราจำแนกตามทัศนคติ และเครื่องหมายดังกล่าวช่วยจำแนกตามทัศนคติได้ จากนั้นจัดคำศัพท์ภายในรายการให้อยู่ในรูปเอกพจน์ ซึ่งนับรวมคำศัพท์และเครื่องหมายได้ 1666 คำ

รายการคำศัพท์และเครื่องหมายที่ได้ในข้างต้นจะถูกนำมาใช้เป็นคุณลักษณะ โดยจะมีค่าเป็น 0 หรือ 1(โดยที่ค่า 0 คือไม่พบในความเห็นนั้น และ ค่า 1 คือพบในความเห็นนั้น) ซึ่งจะขอเรียกเวกเตอร์ของแต่ละความเห็นว่า "อินสแตนซ์" ในงานนี้เราได้ใช้การจำแนกโดย Naïve Bayes ดังสมการที่ 3 ดังนั้นเราจึงใช้อินสแตนซ์ร่วมกับ class ของความเห็นนั้นๆมาคำนวณ Prior Probability เพื่อนำมาหา Target Value ซึ่งเป็น + หรือ - ที่ให้ Posterior Probability ที่สูงสุดแก่อินสแตนซ์ใหม่ๆ

นอกจากนั้นในงานนี้ได้เพิ่มงานในขั้นการเตรียมข้อมูลโดยจะคัดเลือกคุณลักษณะก่อนที่จะนำไปทดลองกับ Learning Method ด้วยโดยใช้ Feature Selection Algorithm 2 ชนิด คือ Principal Components Analysis (PCA) [6] และ Relief Algorithm เพื่อเพิ่มการทำงานในช่วงสอนให้มีประสิทธิภาพและมีความถูกต้องมากยิ่งขึ้นโดยที่ PCA จะหาความสัมพันธ์ของโครงสร้างของคุณลักษณะ และลดขนาดของคุณลักษณะลงโดยใช้ Linear Combination ของคุณลักษณะแทน ส่วน Relief จะใช้หลักการทางสถิติ หาค่าคุณลักษณะที่มีความเกี่ยวข้องกับ Target Concept และเรียงลำดับคุณลักษณะที่มีความเกี่ยวข้องกับ Target Concept จากมากไปน้อย

นอกเหนือจาก Naïve Bayes Classifier แล้ว ในงานนี้ได้ใช้ Learning Method อื่นๆมาทดสอบด้วย อันได้แก่ Decision Tree ซึ่งฟังก์ชั่นการเรียนรู้ถูกนำเสนอโดยใช้ต้นไม้ตัดสินใจ และ Multi-Layer Perceptron ซึ่งเป็น Artificial Neural Network ชนิดหนึ่ง

เพื่อการประเมินความถูกต้องของการจำแนกความเห็นตามทัศนคติ ได้ใช้การทดสอบแบบ 5-Fold Cross-Validation ซึ่งทำการแบ่งข้อมูลออกเป็น 5 ส่วนเท่าๆกันแล้ว แล้วใช้ข้อมูลส่วนแรกมาเป็นชุดทดสอบ และอีก 4 ส่วนที่เหลือใช้เป็นข้อมูลสอน เมื่อสอนเสร็จแล้วจึงนำส่วนแรกซึ่งใช้เป็นชุดทดสอบมาทดสอบความถูกต้องในการจำแนก จากนั้นจะทำการสอนและทดสอบครั้งที่ 2 โดยให้ข้อมูลส่วนที่ 2 เป็นชุดทดสอบ และอีก 4 ส่วนที่เหลือใช้เป็นข้อมูลสอน จากนั้นจึงนำข้อมูลชุดที่ 2 มาทดสอบความถูกต้องของการจำแนก จากนั้นจะเปลี่ยนไปใช้ข้อมูลชุดที่ 3, 4 และ 5 มาเป็นชุดทดสอบตามลำดับซึ่งในแต่ละครั้งจะทำการสอนและทดสอบความถูกต้องเช่นเดียวกับครั้งที่ 1 และ 2 ดังที่ได้กล่าวไปแล้ว จากนั้นจึงนำเอาค่าความถูกต้องของการจำแนกความเห็นตามทัศนคติทั้ง 5 ค่ามาหาค่าเฉลี่ย

4. ผลการทดลอง

ในบทความนี้ได้ทำการเปรียบเทียบประสิทธิภาพของการจำแนกความคิดเห็นตามทัศนคติโดยใช้ Learning Method 3 ชนิดได้แก่ Naïve Bayes, Decision Tree, และ Multi-Layer

Perceptron นอกจากนั้นยังได้ทำการเปรียบเทียบว่าเมื่อเลือกคุณลักษณะโดยการใช้ Relief Algorithm ที่เปอร์เซ็นต์ต่างๆ และการใช้ PCA แล้วจะเพิ่มความถูกต้องในการจำแนกโดย Learning Method ทั้งสามชนิดได้มากน้อยเพียงใด โดยผลการเปรียบเทียบแสดงในตารางที่ 1

โดยที่ Relief algorithm ที่นำมาใช้ ในช่วง Preprocessing step ได้ให้คำศัพท์หรือเครื่องหมายในลำดับต้นๆ เช่น not, don't, filled, taking, error, be, should, required, after, page ตามลำดับ และผู้เขียนได้ใช้คุณลักษณะจาก Relief ที่ 1.125%, 2.5%, 4%, 5%,10%, 20% และ 30% แรกของคุณลักษณะตามลำดับ ซึ่งคิดเป็น 20, 40, 66, 81, 166, 333 และ 500 คุณลักษณะแรกตามลำดับ อนึ่งในการทดลองได้ปรับ Parameter ดังนี้ Multi-Layer Perceptron ปรับ Learning Rate เป็น 0.3 และ Momentum Rate เป็น 0.2 ส่วน Decision Tree (J48) ใช้แบบลดทอนกิ่งลง(Pruning)

จากการทดลองพบว่า Naïve Bayes, Decision Tree (J48) ให้ความถูกต้องในการจำแนกความเห็นตามทัศนคติเป็น 62% และ 58% ตามลำดับ แต่ไม่สามารถจำแนกความเห็นตามทัศนคติโดยใช้ Multi-Layer Perceptron ได้

เนื่องจากข้อจำกัดของโปรแกรมที่มีความจำไม่เพียงพอ เนื่องจากขนาดคุณลักษณะใหญ่เกินไปสำหรับการทำงานด้วย Multi-Layer Perceptron (กรณีใช้ทุกคุณลักษณะ) และเมื่อทำการคัดเลือกคุณลักษณะโดย PCA ร่วมกับ Naïve Bayes, Decision Tree และ Multi-Layer Perceptron ตามลำดับ ให้ความถูกต้องในการจำแนกความเห็นตามทัศนคติเป็น 46%, 58% และ 54% ตามลำดับ

เมื่อเปลี่ยน Feature Selection Algorithm จาก PCA เป็น Relief Algorithm พบว่า เมื่อใช้ Relief ร่วมกับ Naïve Bayes ความถูกต้องในการจำแนกเพิ่มจาก 62% เป็น 68% (สูงสุดเมื่อใช้ 2.5% แรกของคุณลักษณะ) ส่วน Relief เมื่อใช้กับ Decision Tree พบว่าค่าความถูกต้องเพิ่มจาก 58% เป็น 63% (สูงสุดเมื่อใช้ 20% แรกของคุณลักษณะ) และเมื่อใช้ Relief กับ Multi-Layer Perceptron พบว่าค่าความถูกต้องเป็น 64% (สูงสุดเมื่อใช้ 20% แรกของคุณลักษณะ)

ตารางที่ 1 ผลความถูกต้องโดย 5-fold cross-validation

Feature Selection Algorithm	Learning Method		
	Multi-Layer Perceptron	Decision Tree (J48)	Naïve Bayes
No Feature Selection Algorithm	-	58%	62%
PCA	54%	58%	46%
Relief ที่ 1.125%	51%	61%	64%
Relief ที่ 2.5%	62%	61%	**68%**
Relief ที่ 4%	62%	60%	67%
Relief ที่ 5%	62%	56%	67%
Relief ที่ 10%	59%	58%	64%
Relief ที่ 20%	**64%**	**63%**	63%
Relief ที่ 30%	-	62%	61%

สำหรับเวลาที่ใช้ในการสอน พบว่าการทำงานของ Multi-Layer Perceptron ใช้เวลานานสุด ส่วนเวลาในการคัดเลือกคุณลักษณะ พบว่า PCA ใช้เวลานานกว่า Relief โดยยิ่งมีคุณลักษณะจำนวนมากจะยิ่งใช้เวลานานทั้งการสอนและการคัดเลือกคุณลักษณะ

5. บทสรุป

จากการทดลองเมื่อเปรียบเทียบประสิทธิภาพการจำแนกความคิดเห็นตามทัศนคติของ Learning Method พบว่า Naïve Bayes ให้ความถูกต้องของการจำแนกสูงสุด และเมื่อใช้ Feature Selection Algorithm ในช่วง Preprocessing Step ก่อนการสอนพบว่า PCA ไม่ได้ช่วยเพิ่มความถูกต้องของการจำแนกทั้งใน Decision Tree และ Naïve Bayes โดยที่เมื่อใช้ PCA ร่วมกับ Naïve Bayes ผลความถูกต้องในการจำแนกมีค่าลดลง

แต่เมื่อนำ Relief Algorithm มาใช้ในช่วง Preprocessing Step พบว่าเพิ่มความถูกต้องของการจำแนกในทุก Learning Method และให้ผลเพิ่มขึ้นอย่างเด่นชัดเมื่อใช้ Relief ร่วมกับ Naïve Bayes คือมีความถูกต้องถึง 68% ดังนั้น Relief Algorithm ซึ่งนำหลักสถิติมาใช้ในอัลกอริทึมด้วย จึงเป็นวิธีการที่น่าสนใจ ที่จะนำมาประยุกต์ใช้ในการคัดเลือกคุณลักษณะในงานจำแนกตามทัศนคติ และเอกสารต่อไป

เนื่องจากในงานของเราจะพิจารณาทั้งความเห็นโดยแทนด้วยเวกเตอร์ของคุณลักษณะและเข้าสู่ขั้นตอนการเตรียมข้อมูล (Preprocessing Step) และการเรียนรู้โดยใช้ Machine Learning Techniques ตามลำดับ จึงเห็นว่าถ้าเพิ่มการเตรียมข้อมูลโดยคัดเฉพาะประโยคที่เป็นการแสดงความคิดเห็นต่อสินค้านั้น แล้วค่อยเข้าสู่ขั้นตอนการเรียนรู้อาจทำให้จำแนกทัศนคติที่ดี และไม่ดีต่อสินค้าและบริการได้ดียิ่งขึ้น

6. เอกสารอ้างอิง

[1] S. Tan and J. Zhang, An empirical study of sentiment analysis for Chinese documents, Expert Systems with Application 34, 2008, pp.2622-2629.

[2] P.D. Turney, 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, 2002.

[3] B. Pang, L. Lee and S. Vaithyanathan, Thumbs up? Sentiment Classification using Machine Learning Techniques, In Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10, 2002.

[4] K. Kira and L.A. Rendell, A Practical Approach to Feature Selection, In Proceedings of the ninth international workshop on Machine learning, 1992

[5] T.M. Mitchell, Machine Learning, McGraw-Hill international edition, 1997.

[6] D.T. LaRose, Data Mining Methods and Models, Wiley-InterScience, 2006.

Some parts involving this study are accepted to be oral presentation at National Conference on Computer Information Technologies 2011 (CIT 2011) Nakhon Pathom, Thailand on 27-28 January 2011.

BIOGRAPHY

NAME	Miss Titima Kasemsritanawat
DATE OF BIRTH	12 March 1984
PLACE OF BIRTH	Bangkok, Thailand
INSTITUTIONS ATTENDED	Mahidol University, 2003-2008 Bachelor Degree of Science (Medical Science) Mahidol University, 2009-2012 Master of Science (Technology of Information System Management)
PUBLICATION / PRESENTATION	ICSEC 2012 (International Computer Science and Engineering Conference), and CIT 2011 (National Conference on Computer Information Technologies)
HOME ADDRESS	159 Soi Phetkasem 81, Phetkasem 81 Road, Kweang Nong Khaem, Khet Nong Khaem, Bangkok, Thailand, 10160
E-MAIL	mom_love_za@hotmail.com