SENTENCE-BASED SENTIMENT ANALYSIS WITH DOMAIN ADAPTATION CAPABILITY

by
GIZEM GEZICI

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of the requirements for the degree of
Master of Science

Sabancı University
August 2013

SENTENCE-BASED SENTIMENT ANALYSIS WITH DOMAIN ADAPTATION CAPABILITY

Approved by:

Assoc. Prof. Dr. Berrin Yanıkoğlu .....
(Thesis Supervisor)

Assoc. Prof. Dr. Yücel Saygın .....
(Thesis Co-Supervisor)

Assoc. Prof. Dr. Albert Levi .....

Assoc. Prof. Dr. Cem Güneri .....

Asst. Prof. Dr. Hüsnü Yenigün .....

Date of Approval: August 13, 2013

SENTENCE-BASED SENTIMENT ANALYSIS WITH DOMAIN ADAPTATION CAPABILITY

Gizem Gezici

Computer Science and Engineering, MS Thesis, 2013

Thesis Supervisor: Berrin Yanıkoğlu

Keywords: sentiment analysis, opinion mining, domain adaptation, lexicon-based, sentence-based features

Abstract

Sentiment analysis aims to automatically estimate the sentiment in a given text as positive, objective or negative, possibly together with the strength of the sentiment. Polarity lexicons that indicate how positive or negative each term is, are often used as the basis of many sentiment analysis approaches. Domain-specific polarity lexicons are expensive and time-consuming to build; hence, researchers often use a general purpose or domain-independent lexicon as the basis of their analysis.

In this work, we address two sub-tasks in sentiment analysis. We introduce a simple method to adapt a general purpose polarity lexicon to a specific domain. Subsequently, we propose new features to be used in a term polarity based approach to sentiment analysis. We consider different aspects of sentences, such as length, purity, irrealis content, subjectivity, and position within the opinionated text. This analysis is used to find sentences that may convey better information about the overall review polarity. Therefore, our work is also focused on the sentence-based sentiment analysis differently from the other works. Moreover, we worked on two distinct domains, hotel and Twitter with three different systems which are compared with the existing state-of-the-art approaches in the literature.

CÜMLE TEMELLI, FARKLI BAĞLAMLARA ADAPTASYON YETENEĞI OLAN DUYGU ANALIZ SISTEMI

Gizem Gezici

Bilgisayar Bilimi ve Mühendisliği, Yüksek Lisans Tezi, 2013

Tez Danışmanı: Berrin Yanıkoğlu

Anahtar Kelimeler: duygu analizi, düşünce madenciliği, bağlam adaptasyonu, veri sözlüğü temelli, cümle temelli özellikler

Özet

Duygu analizi, verilen bir metnin hissiyatını pozitif, negatif veya objektif olarak otomatik bir biçimde tahmin etmeyi, aynı zamanda da bu hissiyatın derecesini belirlemeyi amaçlar. Her bir kelimenin ne kadar pozitif, ne kadar negatif olduğunu gösteren veri sözlükleri birçok duygu analizi yönteminin de temelini oluşturur. Üzerinde çalışılan bağlama-özel veri sözlüklerini oluşturmak ciddi biçimde zaman alan bir süreç olduğu için, araştırmacılar sıklıkla bağlam-bağımsız veri sözlüklerini tercih ediyorlar.

Biz bu çalışmamızda, hissiyat analizinin iki alt problemine çözüm getirmeye çalışıyoruz. Öncelikle, bağlam-bağımsız veri sözlüğü değerlerini temel alan bir yöntemle yeni makine öğrenimi özellikleri öneriyoruz. Bunları cümlelerin uzunluğunu, cümle içindeki kelimelerin ne kadar tek tipte olduğunu (hepsi pozitif ya da hepsi negatif), cümlenin subjektifliğini, dilek kipi içerip içermediğini ve cümlenin verilen metin içindeki yeri gibi farklı özellikleri de hesaba katarak yapıyoruz. Bu analizi verilen metnin genel hissiyatıyla ilgili daha fazla bilgi taşıyan cümleleri bulmak için kullanıyoruz. Bu nedenle, yaptığımız bu çalışma diğer çalışmalardan farklı olarak cümle temelli duygu analizi üzerine yoğunlaşıyor. Ayrıca, bu yapılandırdığımız sistemin duygu analizi konusunda ne kadar başarılı olduğunu değerlendirebilmek için sistemi iki farklı bağlam üzerinde çalıştırıp, sonuçları karşılaştırıyoruz.

This dissertation is dedicated to my parents Şule and Metin Gezici, and my brother, Barış Gezici, for their continous love and support.

ACKNOWLEDGEMENTS

I am deeply indebted to my thesis advisor, Assoc. Prof. Dr. Berrin Yanıkoğlu, for providing me with the opportunity of working with him. This dissertation would not have been possible without his invaluable advice and continous support. The invaluable advice and feedback from Assoc. Prof. Dr. Yücel Saygın shaped this project.

I am very grateful to Dr. Dilek Tapucu for her encouragement and trust that inspired me from the very beginning of this project.

I would also like to mention the kindness and encouragement of my professors. I am especially grateful to Assoc. Prof. Dr. Albert Levi, Assoc. Prof. Dr. Cem Güneri, and Asst. Prof. Dr. Hüsnü Yenigün for agreeing to be on the thesis committee.

In the end, I am very thankful and grateful to my friends, Rahim Dehkharghani, Mus'ab Husaini, and Inanç Arın. Their assistance and the discussions that we made were indispensably helpful on the completion process of this project.

Finally, none of this would have been possible without my family, who has supported and believed me in every situation. I am deeply grateful for their continous love and support.

CONTENTS

LIST OF TABLES

1

INTRODUCTION

Sentiment analysis aims to extract the subjectivity and strength of the opinions indicated in a given text; which together indicate its *semantic orientation*. For instance a given word or sentence in a specific context, or a review about a particular product can be analyzed to determine whether it is objective or subjective, together with the *polarity* of the opinion. The polarity itself can be indicated categorically as positive, objective or negative; or numerically, indicating the the strength of the opinion in a canonical scale.

Automatic extraction of the sentiment can be very useful in analyzing what people think about specific issues or items, by analyzing large collections of textual data sources such as personal blogs, review sites, and social media. Commercial interest to this problem has shown to be strong, with companies showing interest to public opinion about their products; and financial companies offering advice on general economic trend by following the sentiment in social media [43].

Business owners are interested in the feedback of their customers about the products and services provided by businesses. Social media networks and micro-blogs such as Facebook and Twitter play an important role in this area. Micro-blogs allow users share their ideas with others in terms of small sentences; while Facebook updates may indicate an opinion inside a longer text. Automatic sentiment analysis of text collected from social media makes it possible to quantitatively analyze this feedback.

Two main approaches for sentiment analysis are defined in the literature: one approach is called *lexicon-based* and the other is based on *supervised learning*[54]. The lexicon-based approach calculates the semantic orientation of a given text from the polarities of the constituent words or phrases [54], obtained from a lexicon such as the SentiWordNet [18]. In this approach, different features of the text may be extracted from word polarities [52], such as average word polarity or the number of subjective words, but the distinguishing aspect is that there is no supervised learning. Moreover, the text is often treated

as a *bag-of-words*, where the word polarities are extracted over the whole text, without representing word location information. Alternatives to the bag-of-word approach are also possible, where word polarities of the first sentence etc. are calculated separately [68]. Furthermore, as words may have different connotations in different domains(e.g. the word "small" has a positive connotation in cell phone domain; while it is negative in hotel domain), one can use a domain-specific lexicon whenever available. The widely used SentiWordNet [18] and SenticNet [45] are domain-independent lexicons.

Supervised learning approaches use machine learning techniques to establish a model from an available corpus of reviews. The set of sample reviews form the training data from which the model is built. For instance in [44] [64], researchers use the Naive Bayes algorithm to separate positive reviews from negative ones by learning the probability distributions of the considered features in the two classes. Note that in supervised learning approaches, a polarity lexicon may still be used to extract features of the text, such as average word polarity and the number of positive words etc., that are later used in a learning algorithm. Alternatively, in some supervised approaches the lexicon is not needed: for instance in the LDA approach [5][6], a training corpus is used to learn the probability distributions of topic and word occurences in the different categories (e.g. positive or negative sets of reviews) and a new text is classified according to its likelihood of coming from these different distributions.

In this study, we worked on mainly two datasets as TripAdvisor which is composed of hotel reviews [1] and tweets database [34]. Regarding with the hotel dataset [1], we evaluated our two different systems on two different splits of TripAdvisor dataset [1]. Our first system was less complex and were exploiting review properties at word, sentence and review level. The purpose of this work was to investigate mainly the sentence-based features since they have not yet been sufficiently worked on, in the literature. Along with the sentence-based features, we also showed the effects of different type of features on the overall sentiment of a given text. Moreover, we evaluated and compared with the state-of-art approaches even if the TripAdvisor dataset splits were not exactly the same.

Our second system on the other hand was more complex and had two layers. First level was responsible for updating the word polarities obtained from the SentiWordNet [18] which were incompatible with the hotel domain. Second layer was almost the same structure with our first system, the only difference between them is that we have integrated several new features and improved the system. We compared our complex system with Bespalov et al. (2012) [6] since the dataset evaluated on are the same; this dataset was prepared and released by [6]. In comparison to [6], our method is efficient and easy to implement. However, our accuracy is not better than [6] and this is probably because of their complex system which embraces LDA approach.

Additionally, our second system which embraces two layers was evaluated on the tweets database [34], as well which was a totally different database for sure. As imagined, our features mostly work for hotel domain -since we have worked on hotel domain so far- did not work well for tweet database. Therefore, we have included several new features related to emoticons, exclamation and some slang words specifically for the tweet database [34]. However, these features did not become sufficient to achieve a good accuracy for a different dataset -especially if this dataset is composed of tweets- in which people express their emotions and opinions with less words but more smileys. Nevertheless, in the third system, we dealt with two tasks, the first one was to discover the sentiment of a phrase in a specific context and the second one was to obtain the overall sentiment of a tweet. The participated systems were evaluated separetely for these two different tasks.

We have two different systems but our third system already contains the properties of the second system. All of these three systems will be described elaborately in the following sections. Nevertheless, in the Section 5 different result tables will be displayed for different systems and for distinct datasets in order to make a proper comparison.

In this work, we present a supervised learning approach to sentiment analysis, addressing two sub-tasks in sentiment analysis. First, we introduce a simple method to adopt a domain-independent polarity lexicon to a specific domain. The domain-specific lexicon contains the polarity of the words specific to the given domain. We show that even changes in the polarity of a small number of words affect the overall accuracy by a few percent. As a second contribution we propose a sentence-based analysis of the sentiment, using the updated polarity lexicon in feature extraction. While word-level polarities provide a simple yet effective method for estimating a review's polarity, the gap from word-level polarities to review polarity is too big. To bridge this gap, we propose to analyze word-polarities within sentences as an intermediate step. In this way we hope to also address the issue with irrelevant sentences in a given opinion text. Our main approach is based on this two-phase system mentioned above; however we will be describing three systems throughout this study. This is because we had a first system which does not contain the domain-adaptation phase; thus we improved it and came with a two-phase system. Both of these two systems are in hotel domain; whereas our third system works in tweet domain. Distinct datasets did not result in a huge difference in the structure of our systems; yet we made small modifications in order to achieve a sufficiently good accuracy for all systems that we developed so far. Therefore, each of these three systems will be explained in more detail.

The remainder of the thesis is organized as follows. Ch. 2 provides an overview of the state-of-the-art approaches. Ch. 3 proposes our three systems mainly by pointing out the contribution of a domain-specific lexicon and describing our sentence based sentiment

analysis tool. Ch. 4 gives a complete picture of the classification processes for all of the three systems developed so far. Ch. 5 presents the results of several experiments that show that our two-staged approach works well on hotel domain in comparison to the existing state-of-the-art systems and for Twitter domain in which the system should be tuned more. Finally, in Ch. 6 we conclude and outline our ideas for future work.

2

BACKGROUND AND RELATED WORK

An elaborate survey of the previous works for sentiment analysis has been presented in [43]. We will primarily discuss the previous studies on polarity classification from a general perspective. The fundamental approaches classify the polarity of an opinionated text at either the word, sentence or paragraph, or document levels. At document level polarity classification, one may simply think to relate the overall sentiment of a given text to the sentiment of the keywords in it. However, according to an early study by Pang et al. [44] on movie dataset, suggesting these keywords is not an easy task. The pilot study of Pang et al. [44] revealed the difficulty of the document level sentiment polarity classification. Therefore, it is necessary to come up with more intelligent ideas in order to obtain the overall sentiment of a given text accurately.

2.1. Subjectivity

To suggest these intelligent ideas, specific review properties should be exploited. One of the most important review properties which is highly correlated to the overall sentiment is subjectivity. In determining a texts subjectivity, we seek to identify subjective information within an entire text, or, better yet, distinguish which specific parts are subjective. This subjectivity analysis will most likely result in a more accurate overall sentiment estimation. Mihalcea et al. [46] boils down the consequence of several projects on subsentential analysis [4], [17], [53], [61] into the statement that the problem of separating subjective versus objective has often affirmed to be more difficult than the polarity classification. Therefore, advancements in subjectivity classification may presumably lead to influence sentiment classification positively.

Owing to the importance of subjectivity in sentiment classification, there were several attempts to capture the clues of subjectivity. Early study by Hatzivassiloglou and Wiebe [59] investigated the impacts of adjective orientation and gradability on sentence subjectivity. The goal behind this approach was to determine whether a given sentence is subjective or not, by examining the adjectives in that sentence. Sentence-level or subsentence level detection in different domains were the focus of several studies [39], [29], [42], [47], [58], [28], [61], [65]. Wiebe et al. [27] introduces a broad survey of subjectivity recognition using various features and clues.

2.2. Opinion Strength

Apart from the issue of discovering the subjectivity hints, determining the opinion strength is another problem to be addressed. Wilson et al. [51] raise the question of obtaining clause-level opinion strength. It is also noteworthy to mention here is that identifying opinion strength and rating estimation are distinguished from each other. Besides, classifying an opinionated text as neutral (mid-scored) does not mean that the given text is objective (lack of opinion). This is because, one can have an opinion which is neither positive nor negative but in the middle, i.e., mediocre, or so-so, that can be considered as neutral. This is one of the struggles in obtaining opinion strength, or even estimating the rating, especially when one does not want a binary but 4-class classification, e.g. positive, negative, neutral and objective. Since this is a difficult task, recent work also examined relations between word sense disambiguation and subjectivity [57] in order to extract sufficient information for a more accurate sentiment classification.

2.3. Genre Classification

In addition to the subjectivity clues and opinion strength, there is a high probability that subjectivity detection is related to genre classification, as well. For instance, Yu and Hatzivassiloglou [64] on a particular corpus of Wall Street Journal articles accomplished high accuracy (97 %) with a Naive Bayes classifier, where the task is to differentiate articles under *News and Business* (facts) from articles under *Editorial and Letter to the Editor* (opinions). Based on the possible relation between subjectivity and genre, it can be asserted that there may exist a correlation between topic and opinion. As a result, one may consider to explore these two simultaneously; for example, Rilof et al. [48]

discovered that topic-based text filtering and subjectivity filtering are complementary on the experiments in information extraction. Moreover, topic-based filtering may be directly related to overall sentiment estimation in the sense that a given text may contain off-topic parts that may cause incorrect estimation of the overall sentiment of a given text.

2.4. Viewpoints and Perspectives

Along with subjectivity and opinion strength determination, there are several sub-tasks of sentiment classification. There were many studies to analyse sentiment and opinion in political texts and in these tasks, differently from the previous off-topic discussion, researchers focus on general attitudes through the given text instead of opinions about a specific issue or a narrow subject. For example, Grefenstette et al. [22] made an analysis to detect the political orientation of websites by classifying the documents on that site. These type of works can be grouped under the title of ”viewpoints and perspectives”.

2.5. Affect Analysis

Another area, which is related to sentiment classification, is the examination of various affect types, such as the six ”universal” emotions [16]: anger, disgust, fear, happiness, sadness, and surprise [3] [33] [50]. Although there could exist several applications of this type of studies, interesting application in terms of textual sentiment analysis is probably humour recognition in a given text [36]. Humour recognition is challenging to detect without human intervention and therefore, it can easily mislead an automatic system about the overall sentiment of a given opinionated text. Furthermore, based on the discussion about subjectivity so far, a relation can be constructed between the studies about affects and emotions and learning the subjective language task [27].

2.6. Keywords and Position Information

On the light of the discussions about fundamental approaches in sentiment analysis, we can also discuss the various ways that the previous studies exploited the properties of a given text. The traditional approach in information retrieval to denote a piece of text as a feature vector in which each entry corresponds to an individual term. The features in this vector can be computed based on the properties of a given text that are desired to be exploited. The issue of which properties should be exploited is relevant to the method fulfilled in that paper, since various approaches will probably favour different properties. Nevertheless, there have been some previously suggested features and these features have been utilized prevalently. For instance, term frequencies have customarily been crucial in standard IR, as the reputation of tf-idf weighting indicates.

On a related note *hapax legomena*, denoting words that occur once in a given corpus, has been discovered to be high-precision signs of subjectivity [316]. As an example, Yang et al. [63] explored rare terms that the pre-existing dictionary does not contain, like the novel versions of words such as ”bugfested”. The motivation behind this approach is that such words might be related to emphasis and thus subjectivity in blogs.

Other than the selection of the specific words that may be significant for a given text, the position information of words can be important, as well. The position of a token within a textual unit (e.g., in the middle vs. near the end of the given text) can affect the degree to which that token influences the overall sentiment or subjectivity status of that textual unit. Hence, position information is also sometimes represented in the feature vector of the given textual unit [30] [44]. In connection with the significance of position information, there is another debate about whether higher-order n-grams are beneficial features or not. For instance, Pang et al. [44] declared that unigrams work better than bigrams for sentiment classification task on movie dataset; whereas Dave at al. [11] discovered that in some circumstances, bigrams and trigrams produce better results for product-review polarity classification. As a result, it can be defended that which order of n-grams yield better results is highly dependent on the corpus that is employed.

2.7. Part of Speech Information

Alternatively, Part of Speech (POS) information is frequently utilized in sentiment analysis and opinion mining. POS information is exploited to overcome word sense disambiguation probably [60]. Additionally, based on previous works it can be asserted that

different word types influence overall sentiment estimation process at different levels. For instance, adjectives have been suggested as features by several researchers [37] [56]. One of the earliest preliminaries for the data-driven estimation of semantic orientation of words was developed for adjectives [25]. Subsequently, there was a study on subjectivity detection which showed a high correlation between the presence of adjectives and sentence subjectivity [26]. This discovery has often been taken as a proof that (certain) adjectives are good signs of sentiment and hence a number of approaches concentrate on the presence or polarity of adjectives when trying to determine subjectivity of rating of textual units, especially in unsupervised learning. Rather than concentrating on isolated adjectives, Turney [54] suggested to obtain document sentiment based on chosen phrases, in which the phrases are selected by several pre-defined POS patterns, most containing an adjective or adverb.

Nonetheless, the fact that the presence of adjectives in a textual unit most likely affects the subjectivity of that textual unit does not imply that other POS parts have zero contribution. To illustrate this, a study carried out by Pang et al. [44] on movie corpus compared the sentiment classification results with only using adjectives versus using the same number of most frequent unigrams as features. The results with only adjectives were much worse than the unigrams which means that other POS tags also have an impact on sentiment. In concern with the finding of the paper, the researchers declare that nouns (e.g. ”gem”) and verbs (”love”) can be strong signs for sentiment. For example, Riloff et al. [47] particularly examined the extraction of subjective nouns (e.g. ”concern”, ”hope”) via bootstrapping.

2.8. Syntactic Relations

Other than these methods discussed so far, there have been some studies at including syntactic relations in feature sets. Specifically, it seems that short pieces of text is under consideration for such a deeper linguistic analysis. For instance, Kudo and Matsumoto [31] declared that for two sentence-level classification tasks, sentiment polarity classification and *modality identification* (”opinion”, ”assertion” or ”description”), a subtree-based boosting algorithm using dependency-tree-based worked better than the bag-of-words baseline (although there were no considerable difference in comparison to using n-gram-based features).

2.9. Negation

Negation is an additional area of concern in sentiment analysis. The approaches that are related to negation can be considered as supplementary methods in order to estimate the overall sentiment more accurately. The study done by Das and Chen [10] suggested to resolve the negation problem by encoding it, for instance in the sentence of ”I don’t like deadlines” the token ”like” is transformed into the new token ”like-NOT”. However, this is not a complete solution since not all of the explicit negations reverse the polarity of the enclosing sentence. For example, the polarity of the word ”best” should not be reversed in the sentence, ”No wonder this is considered one of the best.”. Na et al [38] examined to model negation more accurately. They focus on particular POS patterns (in which these patterns are different for distinguished negation words), and tag the whole phrase as a negation phrase. For their product reviews on electronics, they achieved to get about 3% improvement in accuracy with this model of negation. Another challenge with negation is that negation can often be used in more vague ways such as ”avoid”, these kind of words cause implicit negation and they can be easily overlooked. Wilson et. al [61] reported other complex negation effects.

2.10. Topic Information

Last but not least, there seems to be a correlation between topic and sentiment in opinion mining. For instance, in a hypothetical article on Wal-mart, the sentences ”Wal-mart reports profits rose” and ”Target reports that profits rose” could show news which bear different types of sentiments (good vs. bad) relating to the subject of the document, Wal-mart [23]. Thus, topic information can be included in features to some extent.

2.11. Our contribution

In the literature, word-based and review-based features have been already proposed for sentiment analysis. However, sentence-based features have not yet been investigated sufficiently. Since this leads to a gap between word-based and reviews-based features, our aim is to bridge this gap with sentence-based features. Moreover, we adopted the idea

proposed by Demiroz et. al. (2012) and updated the domain-independent lexicon, then we seeked the effect of the adapted lexicon to the results of the whole system. Owing to these, we hope to obtain better results on estimating the overall sentiment, as well.

3

APPROACH

Our main system has two main parts: (a) domain-adaptation of a general purpose polarity lexicon and (b) sentiment analysis using the adapted lexicon and new, sentence-based features. We explain these two parts in Section 3.2.2 and 3.2.3, respectively.

For domain-adaptation of a general purpose lexicon, we propose several variations of a simple method which is based on the delta tf-idf concept [35]. We have previously shown the benefits of using the adaptation technique independently [14], by using a simple sentiment analysis algorithm with and without domain adaptation of the used lexicon. In this paper, we use the adapted lexicon as the base, in feature extraction.

For evaluating the document sentiment, we propose some new and sentence-based features based on the word polarities obtained from the adapted lexicon. Our state-of-the-art results on estimating overall document sentiment in two different domains, reported in Section 5.3.4, show the effectiveness of the proposed method.

3.1. Sentence-Based Opinion Miner

Our first system embraces the second phase of the main approach mentioned in Section 3.2.3. However, it is not exactly the same phase since this system is not as complicated as the second one; it does not have a domain-adaptation capability. Therefore, our first system uses the polarity values from the lexicon namely SentiWordNet [18] without updating the polarity values according to the domain that is worked on.

We decided on the features of our first system in order to exploit the properties of hotel reviews. Differently from the existing approaches, this system contains also sentence-based features which have not yet been investigated sufficiently so far. By using these

feature, the fundamental purpose of this system is to seek to find the influence of the sentence-based features to the overall accuracy of the whole system. Nonetheless, we also looked at the effect of each different sentence type separately (i.e. using the features of a specific sentence type only such as subjective, pure sentences etc.). In addition to these, we also compared our complete first system with the existing state-of-the-art approaches. All of the evaluation results of our first system can be found in Section 5.1.

Our second system is a more complex version of the first system which also has the domain-adaptation capability. Nevertheless, it suggests the same features as the first system but uses the updated polarities from the SentiWordNet [18]. Since the SentiWordNet [18] is a domain-independent lexicon, it may not provide sufficiently good results for specific domains. After the evaluation results of the first system on hotel domain, we needed an updated version of the same lexicon based on the study [14] for better results. Subsequently, we integrated the domain-adaptation capability to our system and established the second system which will be explained in the next section.

3.2. Sentence-Based Opinion Miner with Domain Adaptation Capability

Our second system is composed of two phases: (i) The domain adaptation of the lexicon, namely SentiWordNet and (ii) The computation of the proposed features with the domain-adapted lexicon. The system details can be found in detail througout this section.

3.2.1. SentiWordNet

The polarity lexicon we use as the domain-independent lexicon is the SentiWordNet that consists of a list of words with their POS tags and three associated polarity scores $< pol^{-}, pol^{=}, pol^{+} >$ for each word [18]. The polarity scores indicate the measure of negativity, objectivity and positivity, and they sum up to 1. Some sample scores are provided in Table 3.1 from SentiWordNet. Please note that JJ abbreviation stands for adjective, RB for adverb, NN for noun and VB for verb. These are the mainly used sentiment bearing word types.

Table 3.1: Sample Entries from SentiWordNet

Word	Type	Negative	Objective	Positive
sufficient	JJ	0.75	0.125	0.125
comfy	JJ	0.75	0.25	0.0
moldy	JJ	0.375	0.625	0.0
joke	NN	0.19	0.28	0.53
fireplace	NN	0.0	1.0	0.0
failed	VBD	0.28	0.72	0.0

As many other researchers have done, we simply select the dominant polarity of a word as its polarity and use the sign to indicate the polarity direction. The dominant polarity of a word w, denoted by $Pol(w)$, is calculated as:

$$Pol(w) = \begin{cases} 0 & \text{if } \ max(pol^{=}, pol^{+}, pol^{-}) = pol^{=} \\ pol^{+} & \text{else if } \ pol^{+} \geq pol^{-} \\ -pol^{-} & \text{otherwise} \end{cases} \tag{3.1}$$

In other words, given the polarity triplet $< pol^{-}, pol^{=}, pol^{+} >$ for a word w, if the objective polarity is the maximum of the polarity scores, then the dominant polarity is 0. Otherwise, the dominant polarity is the maximum of the positive and negative polarity scores where pol^{-} becomes $-pol^{-}$ in the average polarity calculation. For example, the polarity triplet of the word "sufficient" is <0.75,0.125,0.125>; hence Pol("sufficient") = -0.75. Similarly, the polarity triplet of the word "moldy" is <0.375,0.625,0.0>; hence Pol("moldy") = 0.

An alternative way for calculating dominant polarity could be to completely ignore the objective polarity $pol^{=}$ and determine the $Pol(w_i)$ of the word to be the maximum of pol^{-} and pol^{+}. With this method, the dominant polarity of the word "moldy" would be -0.375 instead of 0. However, we preferred the first approach as more appropriate, since many words appear as objective or dominantly objective in SentiWordNet.

3.2.2. Adapting a Domain-Independent Lexicon

The basic idea for domain adaptation is to learn the domain-specific polarities from labeled reviews in a given domain. In order to do that, we analyze the occurrence of the words in the lexicon in positive and negative reviews in a given domain. If a particular word occurs significantly more in positive reviews than in negative reviews, then we assume that this word should have positive polarity for this domain, and vice versa. For instance if a word's dominant polarity is negative, but it occurs very often in positive reviews and not very often in negative ones, we update its dominant polarity.

We propose a couple of alternatives for the update mechanism of a word's polarity. The proposed approaches allow us to adapt a domain-independent lexicon such as SentiWordNet for a specific domain, by updating the polarities of only a small subset of the words. However, we also show that this small set of updated words has a significant contribution to sentiment analysis accuracy. While any domain-independent polarity exicon can be used, we have evaluated our proposed method on the commonly used SentiWordNet. Results with bigger and better lexicons such as SenticNet [45] are expected to be similar, albeit possibly showing smaller benefits.

In order to see which words in the domain appear more in a particular class of reviews, compared to the other class, we first compute the tf-idf (term frequency - inverse document frequency) scores of each word separately for positive and negative review classes. The $tf(w, c)$ counts the occurrence of word w in class c, while $idf(w)$ is the proportion of

documents where the word w occurs, discounting very frequently occurring words in the whole database (e.g. 'not', 'be') [49]. There are quite a few variants of tf-idf computations [41], and the tf-idf variant we use is denoted as $tf.idf$ and computed as:

$$tf.idf(w_i, +) = tf(w_i, +) \times idf(w_i) = log_e(tf(w_i, +) + 1) \times log_e(N/df(w_i)) \tag{3.2}$$

$$tf.idf(w_i, -) = tf(w_i, -) \times idf(w_i) = log_e(tf(w_i, -) + 1) \times log_e(N/df(w_i))$$

where the first term is the scaled term frequency (tf) and the second term is the scaled inverse document frequency (idf). The term $df(w_i)$ indicates the document frequency which is the number of documents in which w_i occurs and N is the total number of documents (reviews in our case) in the database.

We then define a new measure for polarity adaptation of words, called $(\Delta tf)idf$:

$$(\Delta tf)idf(w_i) = [tf(w_i, +) - tf.idf(w_i, -)] \times idf(w_i) = tf.idf(w_i, +) - tf.idf(w_i, -)$$

This measure is used in estimating whether the polarity of a word should be adjusted, considering its occurrence in positive and negative reviews separately.

Our new measure is similar to the *Delta TFIDF* term defined in [35] for calculating the polarity scores of words. As shown in Eq. 4, *Delta TFIDF*(w_i, d) score of a word w_i in document d considers the difference in the document frequencies of that word in positive and negative corpora. Then, these scores are summed for each word in document d, to obtain a sentiment value for the document.

In contrast, $(\Delta tf)idf(w_i)$ of word w_i considers the difference between the *term* frequencies of the word w_i in positive and negative reviews.

$$\text{Delta TFIDF}(w_i, d) = tf(w_i, d) \times [idf(w_i, +) - idf(w_i, -)]$$

In this process we excluded words with POS tags containing "PRP" or "DT" to exclude stop words such as "the","I","a", etc.

Last but not least, there are some word-POSTag entries occur more than once in the SentiWordNet [18]. This is stemmed from the fact that some entries may bear different sentiments in different contexts. To deal with these entries, ideally word-sense disambiguation should be considered. However, word-sense disambiguation is a distinct research area, and therefore we did not integrate it to our systems.

Apart from this, we update the polarities of some words in the SentiWordNet, for those words we use the updated polarities for our systems in which word-sense disambiguation is not an issue anyway. For the word-sense entries with non-updated polarities, word-sense disambiguation may be an issue and affect the results; however we did not include it to our systems for now.

3.2.3. Sentence Based Sentiment Analysis Tool

For sentiment analysis of a given document or review, we propose and evaluate new features to be used in a word polarity based approach to sentiment classification.

Our approach depends on the existence of a sentiment lexicon that provide information about the semantic orientation of single or multiple terms. Specifically, we use the SentiWordNet [18] where for each term at a specific function, its positive, negative or neutral appraisal strength is indicated (e.g. "good,ADJ, 0.5)

We define an extensive set of 22 features that can be grouped in five categories: (1) basic features, (2) features based on the occurrence of subjective words, (3) delta-tf-idf weighting of word polarities, (4) punctuation, and (5) sentence-level features. These features are listed in Table 3.2 as a summary.

Table 3.2: Summary of Features

Group Name	Feature	Name
Basic	F_1	Average review polarity
	F_2	Review purity
	F_3	Review subjectivity
Seed Words Statistics	F_4	Freq. of seed words
	F_5	Avg. polarity of seed words
	F_6	Stdev. of polarities of seed words
Δ TF*IDF	F_7	Δ TF*IDF scores of subj. words
	F_8	Δ TF*IDF weighted avg. polarity of subj. words
Punctuation	F_9	# of Exclamation marks
	F_{10}	# of Question marks
	F_{11}	Number of positive smileys
	F_{12}	Number of negative smileys
Sentence Level	F_{13}	Avg. First Line Polarity
	F_{14}	Avg. Last Line Polarity
	F_{15}	First Line Purity
	F_{16}	Last Line Purity
	F_{17}	Avg. pol. of subj. sentences
	F_{18}	Avg. pol. of pure sentences
	F_{19}	Avg. pol. of non-irrealis sentences
	F_{20}	$\Delta TF * IDF$ weighted polarity of first line
	F_{21}	$\Delta TF * IDF$ scores of subj. words in the first line
	F_{22}	Number of sentences in review

3.2.4. Notation

A review R is a sequence of sentences $R = S_1 S_2 S_3 ... S_M$ where M is the number of sentences in R. The review R is also viewed as a sequence of words $w_1 .. w_T$, where T is the total number of words in the review.

3.2.5. Basic Features

As the main features, we use review polarity, purity and subjectivity, which are commonly used in sentiment analysis. In our formulation $pol(w_j)$ denotes the dominant polarity of w_j of R, as obtained from SentiWordNet, and $|pol(w_j)|$ denotes the absolute polarity of w_j.

$$\text{Average review polarity} = \frac{1}{T} \sum_{j=1..T} pol(w_j) \tag{3.3}$$

$$\text{Review purity}(\sum_{j=1..T} pol(w_j))(\sum_{j=1..T} |pol(w_j)|) \tag{3.4}$$

For a sentence $S_i \in R$, the average sentence polarity is used to determine subjectivity of that sentence. If it is above a threshold, we consider the sentence as subjective, and include it in the set of subjective sentences in the review ($subjS(R)$).

3.2.6. Seed Word Statistics

Like some other researchers, we also use a smaller subset of the lexicon, SentiWordNet, consisting of obviously 20 positive and 20 negative seed words, with the hope that they can be more indicative of a reviews's polarity and help the system on estimating the overall sentiment.

The set of seed words ($SeedW$) is defined as the most 20 positive and 20 negative words according to the $\Delta tf * idf$ scores computed for domain-adaptation purpose on the training data, they are shown in Table 3.3 and 3.4 below.

Furthermore, $SeedW(R)$ is defined as the seed words in $SeedW$ that appear most frequently in review R. The motivation behind this is to capture highly positive and negative for a specific corpus that is being worked on.

$$\text{Freq. of subjective words} = |SeedW(R)|/|R| \tag{3.5}$$

$$\text{Avg. polarity of subj. words} = \frac{1}{|SeedW(R)|} \sum_{w_j \in SeedW(R)} pol(w_j) \tag{3.6}$$

Table 3.3: Sample Positive Seed Words

Word	POSTag
great	JJ
excellent	JJ
wonderful	JJ
perfect	JJ
comfortable	JJ
clean	JJ

Table 3.4: Sample Negative Seed Words

Word	POSTag
worst	JJ
dirty	JJ
terrible	JJ
awful	JJ
noisy	JJ
uncomfortable	JJ

As you can see the Table 3.3 and 3.4, most of the seed words are adjective (JJ) which is very expected since most of the time the sentiment words are adjective. Also, most of the time these are the words in a review that bear the overall sentiment of the review.

3.2.7. Δ*tf***idf* Features

We compute the $\Delta tf * idf$ scores of the words in SentiWordNet from a training corpus in the given domain, in order to capture domain specificity [35]. For a word w_i, $\Delta tf * idf(w_i)$ is defined as

$$\Delta tf * idf(w_i) = tf * idf(w_i, +) - tf * idf(w_i, -) \tag{3.7}$$

If the $\Delta tf * idf$ score is positive, it indicates that a word is more associated with the positive class and vice versa, if negative. We computed these scores on the training set which is balanced in the number of positive and negative reviews.

Then, as a feature (f_7), we sum up the $\Delta tf * idf$ scores of all subjective words ($SubjW$). By doing this, our goal is to capture the difference in the distribution of these words, among positive and negative reviews. The aim is to obtain context-dependent scores that may replace the polarities coming from SentiWordNet which is a context-independent lexicon [18]. With the help of context-dependent information provided by $\Delta tf*idf$ related features, we expect to better differentiate the positive reviews from negative ones. As another feature, we tried combining the two information, where we weighted the polarities of all words in the review by their $\Delta tf * idf$ scores (F_8).

3.2.8. Punctuation Features

We have four features related to punctuation. Two of these features were suggested in [15] and since we have seen that they could be useful for some cases, we included them as well as the smileys in our sentiment classification system [8][40].

3.2.9. Sentence Level Features

Sentence level features are extracted from some specific types of sentences that are identified through a sentence level analysis of the corpus. For instance the first and last lines polarity/purity are features that depend on sentence position; while average polarity of words in subjective, pure and irrealis sentences are new features that consider features of subjective, pure or irrealis sentences, respectively.

Subjective sentences are defined in Section 3.5. Similarly, we consider a sentence S_i as *pure* if its purity is greater than a fixed threshold τ. Sentence purity can be calculated as in Eq. 4, using only the words in the sentnece. We experimented with different values of τ and for evaluation we used $\tau = 0.8$.

We also looked at sentences containing *irrealis* words, in order to discount the polarity calculated from those sentences. In order to determine irrealis sentences, the existence of the modal verbs 'would', 'could', or 'should' is checked. If one of these modal verbs appear in the sentence then these sentences are labeled as irrealis similar to [52]. These three sets are called $subS(R)$, $pureS(R)$ and $nonIrS(R)$ in Table 3.5.

Table 3.5: Sentence-Level Features for a review R

F_{14}	Avg. First Line Polarity	$\frac{1}{S_1}\sum_{w\in S_1 pol(w)}$
F_{15}	Avg. Last Line Polarity	$\frac{1}{S_M}\sum_{w\in S_M} pol(w)$
F_{16}	First Line Purity	$[\sum_{w\in S_1} pol(w)]/[\sum_{w\in S_1} \|pol(w)\|]$
F_{17}	Last Line Purity	$[\sum_{w\in S_M} pol(w)]/[\sum_{w\in S_M} \|pol(w)\|]$
F_{18}	Avg. pol. of subj. sentences	$\frac{1}{\|subjS(R)\|}\sum_{w\in subjW(R)} pol(w)$
F_{19}	Avg. pol. of pure sentences	$\frac{1}{\|pureS(R)\|}\sum_{w\in pure(R)} pol(w)$
F_{20}	Avg. pol. of non-irrealis sentences	$\frac{1}{\|nonIrS(R)\|}\sum_{w\in nonIr(R)} pol(w)$
F_{21}	Δtf*idf weighted polarity of 1st line	$\sum_{w\in S_1} \Delta tf * idf(w) \times pol(w)$
F_{22}	Δtf*idf Scores of 1st line	$\sum_{w\in S_1} \Delta tf * idf(w)$
F_{23}	Number of sentences in review	M

3.2.10. Sentence Level Analysis for Review Polarity Detection

We tried three different approaches in obtaining the review polarity. In the first approach, each review is pruned to keep only the sentences that are possibly more useful for sentiment analysis. For pruning, thresholds were set separately for each sentence level feature. Sentences with length of at most 12 words are accepted as short and sentences with absolute purity of at least 0.8 are defined as pure sentences. In order to differentiate subjective

sentences, we looked at if a sentence contains at least one subjective word or a smiley if so that sentence is a subjective sentence, otherwise not. For subjectivity of the word, we adopted the same idea that was mentioned in [67].

Pruning sentences in this way resulted in lower accuracy in general, due to loss of information. Thus, in the second approach, the polarities in special sentences (pure, subjective, short or no irrealis) were given higher weights while computing the average word polarity. In effect, other sentences were given lower weight, rather than the more severe pruning.

In the final approach that gave the best results, we used the information extracted from sentence level analysis as features used for training our system. The evaluation results can be found in the results Section 5.2.

3.3. Tweet-Based Opinion Analyzer

Differently from the previous two systems which are evaluated on hotel domain, in our third system we worked on Twitter domain which is a totally distinct domain in terms of structure. Tweets are mostly composed of spoken language while hotel reviews are more structured and close to writing language. Also in Twitter, due to character limitation people try to express more ideas with less words which makes our task even harder. Furthermore, most tweets contain spelling errors, abbreviations etc. which is another issue to deal with.

Based on the reasons mentioned above, Twitter is a difficult domain for the systems that we developed. We first tried our second system on Twitter domain by adding only a simple feature. The feature that we added is called 'review subjectivity' which takes the value of 1 if the review is subjective, otherwise 0. The purpose of adding this feature to our system is the fact that during the experiments we realized that the system had difficulty classifying mostly the neutral tweets. The system decides if the review is subjective or not with the following rule: If a review contains at least one subjective sentence then it is subjective. The subjective sentence definition is the same as the definition above in Section 3.2.10 that our second system embraces.

Apart from the system modifications for Twitter domain, we also did preprocessing to improve the functionality of the parser producing the dependency tree and the POSTag information. We use Stanford NLP Parser (citation ver) to get POSTags (e.g. adjective, noun etc.) and relations between words and for the parser to work properly we tagged usernames and hashtags in Twitter. In addition to tagging, we did several other preprocessing steps which are listed below:

- Find special Twitter tokens such as usertag, hashtag and url and tag them (e.g. #ladygaga is replaced by hashtag).
- Find and replace intensifiers (e.g. goooood became good).
- Find and replace abbreviations (e.g. xoxo became love).

After these steps, we observed that the parser worked better and we could get better results overall since also the abbreviations were replaced by their meaningful longer version. Thereby, the words in a tweet can be found in the lexicon and its polarity value can be obtained.

Nevertheless, as the reader will see in the Section 5 our accuracies are not as expected since our systems were not developed particularly for Twitter. Preprocessing definitely helped the system to capture the domain better. For Twitter domain, we should add more

features which can be useful for this specific domain like ngrams and bigrams and other features that can exploit the information in a tweet better. This is left as future work, the results that we obtained with existing features and preprocessing can be seen in Section 5.3.

4

CLASSIFICATION

Our main task is to classify reviews with user-given labels. We used two different corpora namely TripAdvisor and Twitter. TripAdvisor corpus is composed of hotel reviews which contain review in text and a user-given label from 1-star (most negative) to 5-star (most positive). For this dataset, we have different classification tasks, binary-classification, three-class classification, four-class classification and five-class classification. Moreover, we have a different dataset namely Twitter which is composed of tweets. This dataset contain tweets in text and a label that shows the sentiment of the tweet. To accomplish these distinct classification tasks, we developed three different systems.

In each of these three systems that we used different classification methods. The details of the classification task in each system are described in the following sections.

4.1. Sentence-Based Opinion Miner

In our first study in which we have fundamentally investigated the effect of sentence-based features on a sample set drawn from the TripAdvisor dataset [1]. Initially, we tried several classifiers that are known to work well for classification purposes. Then, according to their performances we decided to use Support Vector Machines (SVM) and Logistic regression. SVMs are known for being able to handle large feature spaces limiting overfitting. Logistic Regression is a simple, and commonly used, well-performing classifier. The SVM was trained using a radial basis function kernel as provided by LibSVM [9]. For LibSVM, RBF kernel worked better in comparison to other kernels on our dataset. Afterwards, we performed grid-search on validation dataset for parameter optimization by using WEKA [62]. In this work, we only did binary classification on hotel reviews.

4.2. Sentence-Based Opinion Miner with Domain Adaptation Capability

The second system had two-layered structure and was evaluated on a sample set drawn from TripAdvisor dataset which was prepared and released by [6]. We trained our system on the training dataset and obtained a classification model. Then we tested this classificayion model of the system on our test data in order to get the generalization performance. We used LibSVM package in WEKA 3.6 [62] for train-test phase. We did parameter optimization for the kernel, cost and gamma parameters of LibSVM on validation set by using WEKA 3.6 [62]. For kernel, we tried RBF & linear kernel and observed that RBF kernel worked better than linear kernel for our task. For classification, we used C-SVC (classification), RBF kernel and the best parameter pair obtained by parameter optimization (grid-search).

We took 1-star, 2-star reviews as negative and 4-star, 5-star reviews as positive and did a binary classification for TripAdvisor dataset.

In order to make 3-class, 4-class and 5-class classifications we initially made regression and then rounded the regression values e.g. 1.3 became 1.0; whereas 1.6 became 2.0. Thus, when we made regression and then classification, we obtained two different error metrics which are Mean Absoulute Error (MAE) for regression and accuracy for classification on these tasks, e.g. 3,4 and 5-class classification tasks. We preferred this method in order to compare with the state-of-the-art approaches.

For regression, we used epsilon-SVR (regression) as SVM type and set the normalization to true by default. Subsequently, we again made a parameter optimization for cost and gamma parameters of LibSVM with the help of WEKA 3.6 [62]. Based on the results we obtained for cost and gamma parameters, 10.0 seemed to be the best value for both of these parameters as in the binary classification task. Therefore, we did regression in order to do 3,4 and 5-class classification and achieved a MAE. Then we rounded the values and compared them with the true labels of the reviews and obtained an accuracy value. Afterwards in order to compare our work with the systems of Bespalov et. al. [5] [6], we converted our accuracy value to an error rate which is obtained when the error rate is substracted from 100. By this way, we were able to compare our work with Bespalov et. al. [5] [6].

4.3. Tweet-Based Opinion Analyzer

The third system is the most complex one with preprocessing and more features among the three systems. The third system was evaluated on a twitter dataset described in [34]. There were mainly two tasks, namely TaskA and TaskB and two different datasets, namely twitter and SMS datasets which have almost the same structure. The training dataset was composed of only tweets; however the test dataset contained both the tweets and the SMSs which made the task more difficult. The goal of TaskA was to discover the contextual sentiment of a phrase on the other hand TaskB required the overall sentiment of a tweet or a SMS. We focused on TaskB since our system was more suitable for that task.

We established our third system in which we combined two distinct systems and then extracted features from the given dataset [34]. The extracted features are fed into a Naive Bayes classifier, also chosen for its simplicity and successful use in many problems. We have used WEKA 3.6 [24] implementation for this classifier, where the Kernel estimator parameter was set to true.

First subsystem was another system developed in Dehkharghani et al. (2012). The second subsystem actually is the third system which was described in Section 3.3.

We have two independently developed systems [13][19] that were only slightly adapted for the Twitter dataset; therefore we applied a sophisticated classifier combination technique. Rather than averaging the outputs of the two classifiers, we used the validation set to train a new classifier, in order to learn how to best combine the two systems. Note that in this way the combiner takes into account the different score scales and accuracies of the two sub-systems automatically.

The new classifier takes the probabilities assigned by the systems to the three possible classes (positive, objective, negative) as features and another feature which is an estimate of subjectivity of the tweet or SMS messages. We trained the system using these features obtained from the validation data for which we had the groundtruth, with the goal of predicting the actual class label based on the estimates of the two subsystems.

5

IMPLEMENTATION AND EXPERIMENTAL EVALUATION

In this section, the implementation and experimental results are described for the three different systems described in the previous sections.

5.1. Sentence-Based Opinion Miner

In this section, we provide an evaluation of the sentiment analysis features based on word polarities. We use the dominant polarity for each word (the largest polarity among negative, objective or positive categories) obtained from SentiWordNet [18]. We evaluate the newly proposed features and compare their performance to a baseline system. These newly proposed features exploit different properties of the review (e.g. purity, punctuation etc.). Our baseline system uses two basic features which are the average polarity and purity of the review. These features are previously suggested in [2] and [66], and they are widely used in word polarity-based sentiment analysis.

In the evaluation part for our first opinion miner system, we use two different ways of evaluations: Firstly, we investigate the impact level of different type of features on the overall review sentiment. Secondly, we compare our first system to the state-of-the-art systems. The evaluation procedure we use in our experiments is described elaborately in the following subsections.

5.1.1. Dataset

We evaluated the performance of our system on the TripAdvisor dataset that was introduced by [1] and, [55] respectively. The TripAdvisor corpus consists of around 250.000 customer-supplied reviews of 1850 hotels. Each review is associated with a hotel and a star-rating, 1-star (most negative) to 5-star (most positive), chosen by the customer to indicate his or her evaluation.

We evaluated the performance of our approach on a randomly chosen dataset from TripAdvisor corpus. Our dataset consists of 3000 positive and 3000 negative reviews. After we have chosen 6000 reviews randomly, these reviews were shuffled and split into three groups as train, validation and test sets. Each of these datasets have 1000 positive and 1000 negative reviews.

We computed our features and gave labels to our instances (reviews) according to the customer-given ratings of reviews. If the rating of a review is higher than 2 then it is labeled as positive, and otherwise as negative. These intermediate files were generated with a Java code on Eclipse and given to WEKA [62] for binary classification.

5.1.2. Experimental Results

In order to evaluate our first sentence-based opinion miner, we used binary classification with two classifiers, namely SVM and Logistic Regression. The reviews with star rating higher than 2 are positive reviews and the rest are negative reviews in our case, since we focused on binary classification of reviews. This is the first level of the evaluation for our sentence-based opinion miner.

Subsequently, as a deeper analysis we also seeked to find the importance of the features. The importance of the features was obtained using the feature ranking property of WEKA [62] as well as the gradual accuracy increase, as we add a new feature to the existing subset of features.

Apart from these, as a last evaluation step, since our tool is sentence-based opinion miner and sentences have not been taken into account sufficiently in the literature we looked into the effect of different sentence types to the overall sentiment of a given opinionated text, as well. This is a very crucial part of our first system indeed because different sentence types can be exploited deeply based on the results and this may even lead to an improvement in the accuracy of overall review sentiment.

For these results, we used grid search on validation set. Then, with the optimum parameters, we trained our system on training set and tested it on testing set.

Table 5.1: LibSVM Classifier with Grid-Search, Short Sentences (threshold length of 12) & Purity (threshold 0.8)

Dataset	Accuracy
Baseline	84%
NoIrrealis Sentences	80%
Pure Sentences	82%
Short Sentences	72%
Subjective Sentences	82%

The results in Table 5.1 are important also because it can be seen that sentence-based features are not sufficient alone. This is probably because we lose some information when we include only the features about one sentence type (e.g. pure, short etc.). Nonetheless, if these different sentence type features are included together and also utilized with other type of features they can improve the accuracy of the overall sentiment.

Table 5.2: The Effects of Feature Subsets on TripAdvisor Dataset

Feature Subset	Accuracy (SVM)	Accuracy (Logistic)
Basic (F1,F2)	79.20%	79.35%
Basic (F1,F2) + $\Delta TF * IDF$ (F6,F7)	80.50%	80.30%
Basic (F1,F2) + $\Delta TF * IDF$ (F6,F7) + ... Freq. of Subj. Words (F3)	80.80%	80.05%
Basic (F1,F2) + $\Delta TF * IDF$ (F6,F7) + ... Freq. of Subj. Words (F3) + Punctuation (F8,F9)	80.20%	79.90%
Basic (F1,F2) + $\Delta TF * IDF$ (F6,F7) + ... Occur. of Subj. Words (F3-F5)	80.15%	79.00%
All Features (F1-F19)	**80.85**%	**81.45**%

Regarding the results in Table 5.2, we seek information about the effects of different groups of features. With the help of this, it can be understood that some groups of features are more effective than the others. Then, the effective groups of features can be exploited more and new features related to these groups can be suggested. Thus, we may obtain better results than the results in Table 5.3 on TripAdvisor corpus.

Table 5.3: Comparative Performance of Sentiment Classification System on TripAdvisor Dataset

Previous Work	Dataset	F-measure	Error Rate
Gindl et al (2010) [20]	1800	0.79	-
Bespalov et al (2011) [5]	96000	0.93	7.37
Peter et al (2011) [32]	103000	0.83	-
Grabner et al (2012) [21]	1000	0.61	-
Our System (2012)	6000	0.81	-

5.1.3. Discussion

As can be seen in Table 5.3, using sentence level features bring improvements over the best results, albeit small. This means that even if our sentence-based opinion miner system is highly useful on investigating the effect of sentence-based features mainly, we should integrate additional information to our system in order to improve our opinion miner tool. This directs us to a more improved system, our second tool namely, Sentence-Based Opinion Miner with Domain Adaptation Capability. The additional info that we integrate to our second system is not on exploiting the review more with smarter features but adapting the lexicon according to the domain that is working on. Our second system will be described in more detailed way in the following sections.

5.2. Sentence-Based Opinion Miner with Domain Adaptation Capability

In our second system, we again provide an evaluation of the sentiment analysis features based on word polarities. However, this time our goal is to show the aggreagated effect of domain adapted lexicon and our features previously defined in our first tool. To use a domain-specific lexicon improves the overall accuracy. Yet, building a domain-specific lexicon is a costly process; therefore adapting an already constructed domain-independent lexicon to a domain-specific one was suggested by [14]. As an improvement we included this idea to our first system, described in the Section 5.1, with the hope of achieving a better accuracy on estimating overall review sentiment.

5.2.1. Dataset

We evaluated the performance of our second system on a sample drawn from the, TripAdvisor dataset that was prepared by [7]. The whole TripAdvisor corpus was already described in the Section 5.1.1.

The dataset we used in includes around 90.000 customer-supplied reviews in total. However, this dataset was split into three sub-datasets as train, validation and test by [7]. Train dataset consists of around 76.000 while validation dataset is composed of 6.000 and test dataset includes 13.000 reviews. Each of these three subsets of data contains balanced number of reviews in terms of binary sentiment polarity. Nonetheless, this dataset also includes neutral reviews (e.g. with a rating value of 3) and neutral reviews were sampled separately to these three subsets of data by trying not to lose balance of reviews.

As the reader will probably notice, our first and second tool were evaluated on TripAdvisor corpus [1] and, [55]; however different subsets of data were used for evaluation. We generated the first subsets of data with random operations; whereas the second data split was created by [7] and already used in their work [7]. Furthermore, first dataset was small (6000 reviews in total) in comparison to the second one which is composed of around 90.000 reviews. This is because, with our first tool we fundamentally aimed to investigate the effect of our proposed sentence-based features, even if we gave the comparison results with the existing systems. On the other hand, with our second tool our goal was to establish a more advanced system that can be compared with the state-of-the-art systems meaning that our second system was designed mostly to achieve a better overall accuracy instead of seeking the impact of distinct group of features. Thus, we integrated

the idea of adapting a domain-independent lexicon [14] to our first tool and came up with an improved one namely, *Sentence-Based Opinion Miner with Domain Adaptation Capability*.

5.2.2. Implementation

For the implementation of our system, as a first step we computed $\Delta tf * idf$ scores of the words which have POSTags of noun, adjective, verb and adverb in the training set. $\Delta tf * idf$ score of a word-POSTag entry may give an idea about the dominant sentiment (i.e. positive, negative, or neutral) of that entry takes in a specific domain, in our case this is the hotel domain. We took into account the POSTag as well since a word may have different sentiment tendencies for its different POSTags. As an example, "good" takes neutral sentiment value if its POSTag is noun; whereas its sentiment becomes positive if its POSTag is adjective. This is obviously a general example; word-POSTag entries may have different sentiment values in distinct domains. Nonetheless, the example is quite useful to illustrate that working with only word itself is not sufficient, its POSTag should be taken into account for a proper analysis. Moreover, being noteworthy to mention that the $\Delta tf * idf$ scores of the word-POSTag entries were computed only from the training set. $\Delta tf * idf$ scores of sample words can be found in the Table 5.4 below. Please not that these $\Delta tf * idf$ scores have not yet been normalized to [-1,1] which is the polarity range of the words in SentiWordNet [17].

Table 5.4: $\Delta tf * idf$ Scores of Sample Words on TripAdvisor Corpus

Word	POSTag	$\Delta tf * idf$
horror	NN	-330.01
mistake	NN	-1343.52
close	JJ	374.13
old	JJ	-2730.1
quiet	JJ	-2317.55
great	JJ	7338.35
noise	NN	-1281.31
clean	JJ	-2223.33
helpful	JJ	3158.11
rude	JJ	-3650.27
spacious	JJ	1872.4
worst	JJ	-4452.81
dirty	JJ	-4817.36
terrible	JJ	-3764.13
horrible	JJ	-3512.84
okay	JJ	-1625.21
unfortunately	RB	-1449.22
disappointment	NN	-1371.96
joke	NN	-1353.05
leave	VB	-1316.42
nightmare	NN	-1073.64
avoid	VB	-1493.55

After computing the $\Delta tf * idf$ scores of the word-POSTag entries, we obtained the dominant polarity of these entries (i.e. max polarity value) from SentiWordNet [17]. Then, we compared these two polarity values and checked if there is a disagreement or not. If there is a disagreement, we updated the dominant polarities of the word-POSTag entries obtained from SentiWordNet [17] according to our polarity adaptation procedure described in the related part of the method section. Some of the words that have a disagreement betweent its $\Delta tf * idf$ score and the SentiWordNet [17] dominant polarity are illustrated below. However, we normalized the $\Delta tf * idf$ scores to the range of [-0.9,0.9] instead of [-1,1] in order to normalize the values in a smoother way.

Table 5.5: Sample Disagreement Words on TripAdvisor Corpus

Word	POSTag	$\Delta tf * idf$	Normalized $\Delta tf * idf$ Score	SentiWordNet Polarity
great	JJ	7338.35	0.9	0
comfortable	JJ	3209.87	0.35	0
helpful	JJ	3158.11	0.34	0
friendly	JJ	2963.68	0.32	0
quiet	JJ	2317.55	0.23	0
clean	JJ	2223.33	0.22	0
value	NN	2199.89	0.22	0
easy	JJ	1944.68	0.18	0
modern	JJ	1943.92	0.18	0
specious	JJ	1872.40	0.17	0
large	JJ	1839.66	0.17	0
station	NN	1708.06	0.15	0
fabulous	NN	1695.20	0.15	0
rude	JJ	-3650.27	-0.56	0
however	RB	-2383.64	-0.40	0
smell	NN	-2242.11	-0.38	0
avoid	VB	-1493.55	-0.28	0
joke	NN	-1353.05	-0.26	0.53
aware	JJ	-617.52	-0.16	0.56
refurbishment	NN	-314.29	-0.12	0.63
goodness	NN	-303.97	-0.12	0.81

By doing this, our objective was to determine the word-POSTag entries that took different sentiment values specifically for that corpus and modify their values in a domain-independent lexicon, SentiWordNet [17]. In this manner, we hoped to obtain more accurate sentiment values from the lexicon without establishing a new domain-specific lexicon, which is a costly process. Obviously, domain-specific lexicon most probably works better than an adapted domain-independent lexicon in respond to the cost of the establishing the domain-specific lexicon meaning that there is a trade-off between the cost it requires and the accuracy you get. By using this perspective, we adopted the idea of adapting a domain-independent lexicon [14] and used the updated polarity values for our already proposed features for our first tool. You can find the sample updated words below.

Table 5.6: Sample Updated Words

Word	POSTag	Previous Polarity	New Polarity
great	JJ	0	0.9
comfortable	JJ	0	0.35
helpful	JJ	0	0.34
friendly	JJ	0	0.32
quiet	JJ	0	0.23
clean	JJ	0	0.22
value	NN	0	0.22
easy	JJ	0	0.18
modern	JJ	0	0.18
specious	JJ	0	0.17
large	JJ	0	0.17
station	NN	0	0.15
fabulous	NN	0	0.15
rude	JJ	0	-0.56
however	RB	0	-0.40
smell	NN	0	-0.38
avoid	VB	0	-0.28
joke	NN	0.53	-0.26
aware	JJ	0.56	-0.16
refurbishment	NN	0.63	-0.12
goodness	NN	0.81	-0.12

Subsequently to these steps, we calculated our features with the updated polarity values and generated intermediate files for train, validation and test with actual labels of the reviews. These intermediate files are composed of the feature values of each instance (i.e. review) and were created by a Java implementation on Eclipse environment and given to WEKA [24]. On the Java implementation we also utilized the Stanford NLP Parser [12] in order to obtain POSTags of the words in a review and to generate the dependency tree for the review. Dependency trees allow us to get the relations between the words and help us to extract the sentiment phrases that mostly bear the overall sentiment of the review.

Sample sentiment phrases were extracted and shown in the Table 5.7. Regarding the Table 5.7, dependency relations are given by Stanford NLP Parser [12] It is necessary to mention these dependency relations, as well. The relation of *amod* is used to identify adjective-noun phrases such as 'expensive hotel'. For the noun-noun phrases such as 'hotel experience', *nn* relation and for the adverb-adjective phrases such as 'really nice' *advmod* relation is used. Nonetheless, these relations can be used together if a review contains more complex sentiment phrases such as 'really nice hotel' in which you can find two relations *advmod* for 'really nice' and *amod* for 'nice hotel' part of the review. Lastly, *nsubj* relation is a commonly used relation for our purpose. This relation is nec-

essary basically for such a review: 'Hotel was nice' to be able to understand the adjective 'nice' is related to 'hotel'. The meaning of this review is not different than such a review for instance: 'Nice hotel'. The Table 5.7 contains real sentiment phrases from the TripAdvisor corpus; however we did not include the sentiment phrases with any *nsubj* relation in order not to write the whole sentence (i.e. to be able to show *nsubj* relation property we have to give a whole sentence). Moreover, with Stanford NLP Parser [12], we can obtain if the sentence has a negative word in it with *neg* relation and negate the sentiment of the review (i.e. if the overall sentiment is positive and the review contains *neg* relation then its sentiment would be negated).

Table 5.7: Sample Extracted Sentiment Phrases

Sentiment Phrase	POSTags of the Words	Dependency Relation
horror hotel	JJ-NN	amod
great location	JJ-NN	amod
nice hotel	JJ-NN	amod
fabulous hotel	JJ-NN	amod
very clean	RB-JJ	advmod
horrible rooms	JJ-NN	amod
terrible experience	JJ-NN	amod
fantastic beach	JJ-NN	amod
worst hotel experience	JJ-NN-NN	amod-nn
absolutely beautiful	RB-JJ	advmod
poor customer relations	JJ-NN-NN	amod-nn
not bad hotel	RB-JJ-NN	neg-amod
uncomfortable bed	JJ-NN	amod

5.2.3. Results

After the parameter optimization phase, with the best parameters we got the generalization performance of our system and compared it with the existing systems. In this evaluation part, our aim is to compare our second system with mainly Bespalov et. al. (2011) [5] and Bespalov et. al. (2012) [6]. This is because, our first system [19] is almost already better than the other state-of-the-art approaches except Peter [32]. Nevertheless, we cannot make a proper comparison with Peter et. al. (2001) since they divided the hotel reviews into two classes with different rating values (i.e. rating 1 vs. rating 4&5) which makes the classification process easier. Therefore, our first system is better than all of the existing systems that can be compared with. Thus, there is no need to compare our second system,

which is a more advanced one, with other studies except the systems of Bespalov et. al. [5][6]. Nevertheless, in order to provide a complete review of state-of-the-art approaches we preferred to include the evaluation results of our first system as well.

Table 5.8: Recent Results on the TripAdvisor Corpus

Previous Work	Dataset	F-measure	Error Rate	Task
Peter et al. [32]	103000	0.83	-	Binary: 1 vs. {4,5}
Gindl et al. [20]	1800	0.79	-	Binary: {1,2} vs. {4,5})
First System [19]	6000	0.81	-	Binary: {1,2} vs. {4,5}
Bespalov et al. [5]	96000	-	7.37	Binary: {1,2} vs. {4,5}
Bespalov et al. [6]	96000	-	6.90	Binary: {1,2} vs. {4,5}
This system	96000	-	13.23	Binary: {1,2} vs. {4,5}
Grabner et al. [21]	1000	0.55	-	Three-class: {1,2}, {3}, {4,5}
This work	96000	0.63	36.50	Three-class: {1,2} vs. {3} vs. {4,5}
Bespalov et al. [5]	96000[1]	-	39.60	Four-class: {1}, {2}, {4}, {5}
Bespalov et al. [6] [1]	96000	-	31.41	Four-class: {1}, {2}, {4}, {5}
This system	96000	0.49	50.71	Four-class: {1}, {2}, {4}, {5}
Bespalov et al. [5]	96000	-	49.20	Five-class: {1}, {2}, {3}, {4}, {5}
Bespalov et al. [6]	96000	-	40.76	Five-class: {1}, {2}, {3}, {4}, {5}
This system	96000	0.44	56.25	Five-class: {1}, {2}, {3}, {4}, {5}

[1]These two datasets are the same dataset released by Bespalov et al. (2012)[6] and therefore the results are directly comparable.

After we categorized the results we achieved, additionally in order to compare our system with [6], it was necessary to obtain a 5-class error rate percentage. Thus, after we got the regression values for class labels of reviews from WEKA [24], we rounded them (e.g. 1.8 became 2 while 1.3 became 1) with the round function of Excel. Then by comparing these rounded values and actual class labels of the instances (e.g. reviews), we obtained an error rate percentage of 56.25 on 5-class classification while [7] obtained 40.76. For three class classification task we achieved error rate of 36.50 and F-measure of 0.63 which is significantly higher than Grabner (2012), which is the only system than can be compared to ours for three-classification task. For four class classification task lastly, we obtained an error rate value of 50.71 while Bespalov (2012) got 40.76. With our more advanced system, *Sentence-Based Opinion Miner with Domain Adaptation Capability*, we still could not obtain a comparable result with any of the works of Bespalov (2011 & 2012). There are several reasons for this and they will be discussed elaborately in the discussion section.

On the light of these, we segmented the results based on the division of classes (i.e. positive, negative, neutral or rating of 1, 2 etc.) to be able to make a suitable comparison. Moreover being noteworthy to mention that our dataset, which our second system was evaluated on, is the same dataset with the Bespalov (2012). Afterwards, as you can see in the Table 5.8, we evaluated our system in three different ways. Firstly, we evaluated it for three-class classification by taking rating value of 1 and 2 as negative class and value of 4 and 5 as positive class and lastly including also the neutral reviews with rating value of 3. Secondly our system was evaluated for four-class classification task excluding the neutral reviews and finally for five-class classification task including all of the reviews including also the neutral reviews.

5.2.4. Discussion

As you can see in the Table 5.8, our second system is the best system so far except it the Bespalov's systems [5][6]. This is stemmed from the fact that both of their systems [5][6] use LDA approach which is a complicated approach that utilized topic-modeling. Although [5][6] obtains the best result on a large TripAdvisor dataset, its main drawback is that topic models learned by methods such as LDA requires re-training when a new topic comes. In contrast, our system uses word polarities; therefore it is very simple and fast. For this reason, it is more fair to compare our system with similar systems in the literature.

5.3. Tweet-Based Opinion Analyzer

Apart from the two systems described above, we have another system which is very similar to those systems. Nevertheless, our task is quite different than the previous tasks. In this task, mainly we are trying to achieve a good accuracy on Twitter database not on hotel dataset anymore meaning that our domain has changed although our system did not change a lot. In our third system, we included one more feature to our system. This feature is a simple one that can convey simple information for our system since the tweets are very short and contains the information in an unstructured but a less complex way in comparison to hotel reviews which are generally quite long and convey the information in a more structured way by containing more complex sentence structures. This feature is called 'review subjectivity' that takes values either 0 or 1. If the review is subjective it takes the value of 1, otherwise 0. The way which we decide the review is subjective or not was explained in detail in the method section below.

On the light of these, our task details and the results that we obtained on Twitter dataset described in [34] can be found below.

5.3.1. Two Tasks We Performed

There were two tasks: 1) Task A where the aim was to determine the sentiment of a phrase within the message (contextual polarity) and 2) Task B where the aim was to obtain the overall sentiment of a message (tweet or SMS).

In each task, the classification involves the assignment of one of the three sentiment classes, positive, negative and objective/neutral. There were two different datasets for each task, namely tweet and SMS datasets [34]. Due to the different nature of tweets and SMS and the two tasks (A and B), we in fact considered this as four different tasks.

5.3.2. Dataset

In our third system, we used diverse datasets for two different tasks, namely TaskA and TaskB.

In addition to these, there are fundamentally two datasets: Tweet and SMS but there are two distinct tasks for each dataset; therefore we have four datasets: TaskA Twitter, TaskA SMS, TaskB Twitter and TaskB SMS.

The datasets were as follows:

- TaskA Twitter dataset contains tweet and user ids of tweets; the beginning and end index of phrases; the tweet itself and the sentiment of the phrase.
- TaskA SMS contains sms and user ids of sms messages, begin and end index of the phrases; the sms itself and the sentiment of the phrase.
- TaskB Twitter dataset contains tweet and user ids of tweets; the tweet itself and the sentiment of the tweet.
- TaskB SMS dataset contains sms and user ids of sms messages; the sms itself and the sentiment of the sms.

The sample reviews from training datasets (i.e. TaskA twitter and TaskB twitter) are displayed in the Table 5.9 and 5.10 below.

Table 5.9: TaskA Twitter Dataset

Tweet id	User id	Begin index	End index	Sentiment	Tweet
*	**	0	12	objective	Gas by my house hit $3.39!!!! I'm going to Chapel Hill on Sat. :)
*	**	13	13	positive	Same Tweet Above (STA)
*	**	0	2	objective	Theo Walcott is still shit, watch Rafa and Johnny deal with him on Saturday.
*	**	3	4	negative	(STA)
*	**	5	13	objective	(STA)
*	**	0	0	objective	its not that I'm a GSP fan, i just hate Nick Diaz. can't wait for february.
*	**	1	1	negative	(STA)
*	**	2	6	objective	(STA)
*	**	7	9	negative	(STA)
*	**	10	15	objective	(STA)

Regarding the Table 5.9, we did not give the tweet id and user id since they have nothing to do with the task we are trying to achieve. Nevertheless, for a complete dataset structure that was provided those id columns were also added to the table. As you can see from the sample tweets from the twitter dataset of TaskA in Table 5.9, based on the begin and end index of a given phrase sentiment values changes. In this manner, for TaskA one tweet can contain more than one sentiment because of different phrases it contains. However, unfortunately there was an ambiguity in these given begin and end indices (i.e. spaces, punctuations are involved or not). There were many cases that these indices may result in a problem; thus we could not achieve a proper result for the TaskA. Nevertheless, we could not put the same level of effort with the TaskB for sure due to time constraints.

Table 5.10: TaskB Twitter Dataset

Tweet id	User id	Topic	Sentiment	Tweet
*	**	chapel hill	positive	Gas by my house hit $3.39!!!! I'm going to Chapel Hill on Sat. :)
*	**	rafa	negative	Theo Walcott is still shit, watch Rafa and Johnny deal with him on Saturday.
*	**	nick diaz	negative	its not that I'm a GSP fan, i just hate Nick Diaz. can't wait for february.
*	**	israel	negative	Iranian general says Israel's Iron Dome can't deal with their missiles (keep talking like that and we may end up finding out)
*	**	tehran	objective -OR-neutral	Tehran, Mon Amour: Obama Tried to Establish Ties with the Mullahs http://t.co/TZZzrrKa via @PJMedia_com No Barack Obama - Vote Mitt Romney
*	**	harry	neutral	I sat through this whole movie just for Harry and Ron at christmas. ohlawd
*	**	poland	positive	with J Davlar 11th. Main rivals are team Poland. Hopefully we an make it a successful end to a tough week of training tomorrow.
*	**	happy valentines day	objective	Why is "Happy Valentines Day" trending? It's on the 14th of February not 12th of June smh..

In the Table 5.10, sample tweets of TaskB are illustrated. As you have noticed differently from the TaskA there is also topic info since the overall sentiment of the tweet is highly dependent on the context. Therefore, as mentioned before TaskB is a harder task than the TaskA which is also visible in the evaluation results. In the experimental results part, we achieved a better accuracy for TaskA in comparison to TaskB even without putting much effort.

Apart from this, there is a sample tweet in the Table 5.10 whose sentiment is 'objective-OR-neutral' which seems kind of having an ambigious sentiment. This is because, initially the dataset contained objective as well as neutral tweets. With this dataset, the system should have differentiated the objective tweets from the neutral ones that was an extremely different task. Objective reviews are the ones that have no opinion in it by containing the facts while neutral reviews have opinion but this opinion is neither positive nor negative. Moreover, a neutral review may contain the same level of positivity and

negativity in it. As it can be inferred from these definitions, it is a very difficult task of differentiating objective reviews from neutral ones for an automated system, especially on a tricky dataset such as Twitter. Because of these reasons, dataset was modified and the objective tweets are labelled as 'objective-OR-neutral' meaning that the system can label it as neutral and this does not cause a misclassification error. Thus, the classification problem of the Twitter dataset is a three-class (i.e. positive, negative, neutral) classification problem instead of four task (i.e. positive, negative, neutral and objective).

5.3.3. Different Systems for Diverse Tasks and Datasets

We mainly worked on TaskB (i.e. overall tweet sentiment) where we had some prior experience, and we evaluated almost the same system on TaskA.

As we did not use more outside labelled data (tweets or SMS), we trained our systems on the available training data which consisted only of tweets and evaluated our systems on both tweets and SMS sets. In fact, we separated part of the training data as validation set and comparison of the two subsystems.

We chose only one system for each task, we selected the three systems (SU1, SU2, combined) based on their performance on the validation set. The performances of these systems are summarized in Table 5.11.

Finally, we re-trained the selected system with the full training data that we could obtain, to use all available data.

For the implementation, we used C# for subsystem SU1 and Java & Stanford NLP Parser [12] for subsystem SU2 and WEKA [24] for the classification part for both of the systems.

However, we have two diverse test sets for each task, TaskA Twitter, TaskA SMS for TaskA and TaskB Twitter, TaskB SMS for TaskB. Therefore, we trained our system on TaskA training set and tested on TaskA Twitter got a result and then also tested our system on TaskA SMS and achieved another result. Similarly for TaskB, we repeated the same process and obtained two different results for two different test sets of TaskB.

We ran different systems in order to get those results; subsystem SU1 was selected for TaskA Twitter while subsystem SU2 was chosen for TaskA SMS. Nonetheless, for both of the datasets of TaskB we used our two-staged combined system. As mentioned in the classifier sections of subsystems SU1 and SU2, logistic function was applied for the classification phase of SU1 and Naive Bayes for SU2. Furthermore, logistic function was utilized for the classification part of our combined system. Although subsystems SU1 and SU2 were implemented in different environments, the classification step of both of these systems was handled via WEKA [24].

Last but not least, if we compare the difficulty level of tasks, it can be defended that the TaskB is more difficult than TaskA mainly. This is because in TaskA you should find the sentiment of a specified phrase; whereas in TaskB the concern is to find the overall sentiment in which one should look at different parameters that may affect the overall sentiment. Especially in Twitter, there are more than one idea are conveyed in a sentence since tweets are shorter and more compacted due to character limitation. Furthermore,

because of different datasets (i.e. tweet and SMS) the difficulty levels of the given tasks (i.e. TaskA and TaskB) were also affected. The training set that was described in [34] is composed of only tweets; therefore when the system is trained by the tweet dataset but evaluated on the SMS dataset the task became even harder. This is rooted from the fact that the train and test datasets does not contain the same pattern and therefore the classification model established from the training set (i.e. tweets) may not be sufficient to predict the SMS dataset.

5.3.4. Results

In order to evaluate and compare the performances of our two systems, we separated a portion of the training data as validation set, and kept it separate. Then we trained each system on the training set and tested it on the validation set. These test results are given below.

We obtained 75.60% accuracy on the validation set with subsystem SU1 on TaskA_twitter using logistic regression. For the same dataset, we obtained 70.74% accuracy on the validation set with subsystem SU2 using a Naive Bayes classifier.

For the twitter dataset of TaskB on the other hand, we benefited from our combined system in order to get better results. With this combined system using logistic regression as a classifier, we achieved 64% accuracy on the validation set. The accuracies obtained by the individual subsystems on this task was 63.10% by SU1 and 62.92% by SU2.

In addition to these, we also obtained the same results for the SMS dataset of TaskA and since we did not combine our systems [13] and [19] we did not give the results for that dataset. Moreover, on the twitter and SMS datasets of TaskA we did not obtain different accuracies since the TaskA requires the sentiment of a given phrase in a context instead of an overall sentiment (i.e. TaskB) which is not affected a lot by different train-test datasets (e.g. train dataset is composed of tweets while test dataset is composed of SMS). This is because we did not use Machine Learning techniques for TaskA but rather a simple approach by obtaining the polarity values of the words in a given phrase from the lexicon since we were aware that the context did not affect the sentiment of a given phrase significantly.

Table 5.11: Results on Twitter Dataset

Dataset	System	Accuracy
TaskA Twitter	SU1	75.60%
	SU2	70.74%
TaskB Twitter	SU1	63.10%
	SU2	62.92%
	Combined	64.00%
TaskB SMS	SU1	63.00%
	SU2	62.00%
	Combined	65.00%

5.3.5. Discussion

The accuracy of our third system for different tasks are not very high due to many factors. First of all, both domains (tweets and SMSs) were new to us as we had only worked on review polarity estimation on hotel domain before.

For tweets, the problem is quite difficult due to especially short message length; misspelled words; and lack of domain knowledge (e.g. 'Good Girl, Bad Girl' does not convey a sentiment, rather it is a stage play's name). As for the SMS data, there were no training data for SMSs, so we could not tune or re-train our existing systems, either.

This was our first experience with the Twitter and SMS domains. Given the nature of tweets, we used simple features extracted from term polarities obtained from domain-independent lexicons. In the future, we intend to use more sophisticated algorithms, both in the natural language processing stage, as well as the machine learning algorithms.

6

CONCLUSION

In this section, we will conclude our work and discuss the future work for each of the three systems described throughout this study. To begin with, we will make a conclusion for our first system and then suggest several future works that can be done to improve the system. By this way, we will be able to make a connection to our second system which is the improved version of the first system.

Our first system is the simplest system that we developed so far. This system was mainly developed for the purpose of investigating the influence of sentence based features. In this study, we observed that choosing the features of one sentence type and using only these features were not sufficient to obtain good results. Thus, it is necessary to exploit the sentence-based features together with the word-based and review-based features. In this manner, it can be observed that the sentence-based features are significantly useful on bridging the gap between the word-based and review-based features.

In addition to seeking the effect of sentence-based features to the prediction of the overall sentiment of a review, we also compared the distinct sentence types and measured their effects, as well. For our task, subjective and pure sentences contributed to the estimation of overall sentiment more in comparison to the non-irrealis and short sentences. Nevertheless, we compared our system with the state-of-the-art approaches and obtained comparable results, except [5]. Bespalov (2011) proposes a complex system which embraces LDA; therefore their system requires re-training when a new topic comes due to the working principle of LDA. In response to such a complex system, our system is simple yet efficient which exploits review properties by using word polarities from SentiWordNet [17]. Thus, it is more fair to evaluate our system with the similar systems in the literature.

On the light of these, we noticed that it is necessary to improve our first system for a better generalization of the hotel domain. To be able to capture the domain info better, which is a requirement for better results on hotel domain, it is necessary to establish a domain-specific lexicon. However, it is a time-consuming process; thus we adopted the idea suggested by Demiroz (2012) which is updating the polarities in the domain-independent lexicon.

In our second system, we integrated the notion of updating the SentiWordNet [17] polarities to our first system [14]. In this manner, we constructed a lexicon that has capability of capturing the context info properly. With the help of updated domain-independent lexicon, our target is to make a generalization of the corpus; thereby to obtain better results on hotel domain.

Apart from these, to illustrate the updating process we also displayed the words whose polarities were updated. When the words in Table 5.5 are examined further, interesting sample words can be found. For instance, the word of 'joke' has dominantly a positive meaning in the SentiWordNet [17] which is intuitive. However, the word of 'joke' is mostly used in negative reviews on training data of hotel dataset such as *the complimentary breakfast was a joke*. Therefore it was necessary to update the dominant polarity value of 'joke'; its dominant polarity was altered from positive to negative because of its sentiment tendency is negative in training set.

Owing to our improved system, we achieved better results on a bigger hotel dataset (6000 reviews in the first system vs. 90000 reviews in the second system) and displayed them in the Section 5.2. Nonetheless, our system is significantly better than the existing approaches, still except the systems of Bespalov et al (2011 & 2012). With our first system, we could not achieve comparable results with Bespalov et al. (2011) and improved our system and built the second system in order to obtain comparable results. However, Bespalov et al. (2012) is an improved version of the Bespalov et al. (2011); therefore even with our improved second system we could not achieve comparable results for both of the systems [5][6]. Nevertheless, both of these systems embraces LDA approach which makes the systems complex; thus it is not fair to compare our simple yet efficient systems with them. In that case, our system is the best amongst the current systems which was evaluated on TripAdvisor corpus [1][55].

Based on the discussion so far, differently from the first two systems the third system was evaluated on Twitter domain. Therefore, we slightly modified our system for this domain. We added a new feature which is related to review subjectivity. It is a simple feature; yet it can help the system to differentiate especially the neutral tweets. Moreover, we spent more time on preprocessing step for tweets, since tweet dataset is not a clean dataset (e.g. many spelling errors, abbreviations etc.) in comparison to hotel dataset which we used

in our previous systems. With the help of the preprocessing steps, our goal was to help the parser work properly on getting POSTags (i.e. adjective, noun etc.) and dependency relations between words (i.e. to capture phrases such as 'small room'). If the parser gives accurate information, then this will contribute to obtain better results on Twitter domain.

Regarding the previously discussed ideas, our third system is the same system with our previous two systems. We made small modifications since our third system was evaluated on Twitter domain; thus our third system is almost the same system with the previous systems. For this reason, we could measure the cross-domain generalization capability of our third system, as well. Related to the results in Section 5.3 we achieved on Twitter domain, it can be advocated that our system did not work as expected on tweet dataset described in [34].

7

FUTURE WORK

Based on the conclusions described in the previous section, it is apparent that we need to improve our system in order to obtain better results. We are aware of the fact that the polarities taken from the SentiWordNet [17] are not consistent, especially for a specific domain, namely hotel domain in our case. This stems from the fact that the SentiWordNet [17] is a domain-independent lexicon; thus we adopted the idea of updating the polarities of SentiWordNet [17] which was already proposed in [14]. Instead of establishing a domain-specific one, updating a domain-independent lexicon is a time-saving process. Therefore, we integrated the idea suggested by Demiroz [2012] to the first system and established our second system *Sentence-Based Opinion Miner with Domain Adaptation Capability*.

Although we improved and established a better system, we do not yet have comparable results with the systems of Bespalov et al. (2011 & 2012). Therefore, we need to develop a much better system. Our next step for this is to integrate LDA approach to our second system in order to get comparable results with their systems [5][6].

Besides, we also evaluated our system on a completely different domain, Twitter to evaluate the cross-domain capability of our second system. Thus, we slightly modified our second system for Twitter domain and evaluated on a dataset of tweets.

On the light of the discussions in the Section 5.3.5, it can be stated that our second system which was developed for hotel domain is not good enough for cross-domain tasks. Thus, we decided to improve our system and design it for Twitter domain for better results. This is rooted from the fact that Twitter is a very different domain from the hotel domain. If our system had been evaluated on a movie domain which is similar to hotel domain, we would probably get more comparable results.

For these reasons, as a future work firstly we will evaluate our system on movie domain in order to validate whether we could get comparable results with the existing approaches. Subsequently, we will develop a new system for Twitter domain by adding ngrams and bigrams, which are commonly used features, for Twitter to our feature set.

BIBLIOGRAPHY

[1] The TripAdvisor website. `http://www.tripadvisor.com`, 2011. [TripAdvisor LLC].

[2] Abbasi Ahmed, Chen Hsinchun, and Salem Arab. Sentiment analysis in multiple languages: Feature selection for opinion classification in Web forums. *ACM Transactions on Information Systems*, 26:1–34, 2008.

[3] Cecilia Ovesdotter Alm, Dan Roth, and Richard Sproat. Emotions from text: machine learning for text-based emotion prediction. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 579–586. Association for Computational Linguistics, 2005.

[4] Alina Andreevskaia and Sabine Bergler. Mining wordnet for fuzzy sentiment: Sentiment tag extraction from wordnet glosses. In *Proceedings of EACL*, volume 6, pages 209–216, 2006.

[5] Dmitriy Bespalov, Bing Bai, Yanjun Qi, and Ali Shokoufandeh. Sentiment classification based on supervised latent n-gram analysis. In *ACM Conference on Information and Knowledge Management (CIKM)*, 2011.

[6] Dmitriy Bespalov, Yanjun Qi, Bing Bai, and Ali Shokoufandeh. Sentiment classification with supervised sequence embedding. In *Machine Learning and Knowledge Discovery in Databases*, pages 159–174. Springer, 2012.

[7] Dmitriy Bespalov, Yanjun Qi, Bing Bai, and Ali Shokoufandeh. Sentiment classification with supervised sequence embedding. In *ECML/PKDD (1)*, volume 7523 of *Lecture Notes in Computer Science*, pages 159–174. Springer, 2012.

[8] Albert Bifet and Eibe Frank. Sentiment knowledge discovery in twitter streaming data. In *Discovery Science*, pages 1–15. Springer, 2010.

[9] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines, 2001.

[10] Sanjiv Das and Mike Chen. Yahoo! for amazon: Extracting market sentiment from stock message boards. In *Proceedings of the Asia Pacific Finance Association Annual Conference (APFA)*, volume 35, page 43, 2001.

[11] Kushal Dave, Steve Lawrence, and David M Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web*, pages 519–528. ACM, 2003.

[12] Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D. Manning. Generating typed dependency parses from phrase structure trees. In *LREC*, 2006.

[13] Rahim Dehkharghani, Berrin Yanikoglu, Dilek Tapucu, and Yucel Saygin. Adaptation and use of subjectivity lexicons for domain dependent sentiment classification. In *Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conf.*, pages 669–673, 2012.

[14] Gulsen Demiroz, Berrin Yanikoglu, Dilek Tapucu, and Yucel Saygin. Learning domain-specific polarity lexicons. In *Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conf. on*, pages 674–679, 2012.

[15] Kerstin Denecke. How to assess customer opinions beyond language barriers? In *ICDIM*, pages 430–435. IEEE, 2008.

[16] Paul Ekman and Wallace V Friesen. Felt, false, and miserable smiles. *Journal of nonverbal behavior*, 6(4):238–252, 1982.

[17] Andrea Esuli and Fabrizio Sebastiani. Determining term subjectivity and term orientation for opinion mining. In *Proceedings of EACL*, volume 6, pages 193–200, 2006.

[18] Andrea Esuli and Fabrizio Sebastiani. Sentiwordnet: A publicly available lexical resource for opinion mining. In *In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC06*, pages 417–422, 2006.

[19] Gizem Gezici, Berrin Yanikoglu, Dilek Tapucu, and Yücel Saygın. New features for sentiment analysis: Do sentences matter? In *SDAD 2012 The 1st International Workshop on Sentiment Discovery from Affective Data*, page 5, 2012.

[20] Stefan Gindl, Albert Weichselbraun, and Arno Scharl. Cross-domain contextualization of sentiment lexicons. *Media*, 2010.

[21] Dietmar Grabner, Markus Zanker, Gnther Fliedl, and Matthias Fuchs. Classification of customer reviews based on sentiment analysis. *Social Sciences*, 2012.

[22] Gregory Grefenstette, Yan Qu, James G Shanahan, and David A Evans. Coupling niche browsers and affect analysis for an opinion mining application. In *RIAO*, pages 186–194. Citeseer, 2004.

[23] Bennett A Hagedorn, Massimiliano Ciaramita, and Jordi Atserias. World knowledge in broad-coverage information filtering. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 801–802. ACM, 2007.

[24] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.

[25] Vasileios Hatzivassiloglou and Kathleen R. Mckeown. Predicting the semantic orientation of adjectives. In *Proceedings of ACL-97, 35th Annual Meeting of the Association for Computational Linguistics*, pages 174–181. Association for Computational Linguistics, 1997.

[26] Vasileios Hatzivassiloglou and Janyce M Wiebe. Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the 18th conference on Computational linguistics-Volume 1*, pages 299–305. Association for Computational Linguistics, 2000.

[27] R. Bruce M. Bell J. M. Wiebe, T. Wilson and M. Martin. Learning subjective language. In *Computational Linguistics*, volume 30, page 277308, 2004.

[28] T. Wilson J. M. Wiebe and M. Bell. Identifying collocations for recognizing opinions. In *Proceedings of the ACL/EACL Workshop on Collocation: Computational Extraction, Analysis, and Exploitation*, 2001.

[29] S.-M. Kim and E. Hovy. Automatic detection of opinion bearing words and sentences. In *Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP)*, 2005.

[30] Soo-Min Kim and Eduard Hovy. Automatic identification of pro and con reasons in online reviews. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 483–490. Association for Computational Linguistics, 2006.

[31] Taku Kudo and Yuji Matsumoto. A boosting algorithm for classification of semi-structured text. In *EMNLP*, volume 4, pages 301–308, 2004.

[32] Raymond Yiu Keung Lau, Chun Lam Lai, Peter B. Bruza, and Kam F. Wong. Leveraging web 2.0 data for scalable semi-supervised learning of domain-specific sentiment lexicons. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 2457–2460, New York, NY, USA, 2011. ACM.

[33] Hugo Liu, Henry Lieberman, and Ted Selker. A model of textual affect sensing using real-world knowledge. In *Proceedings of the 8th international conference on Intelligent user interfaces*, pages 125–132. ACM, 2003.

[34] Suresh Manandhar and Deniz Yuret. Semeval tweet competition. In *Proc. of the 7th International Workshop on Semantic Evaluation (SemEval 2013) in conjunction with the Second Joint Conference on Lexical and Comp.Semantics (*SEM 2013)*, 2013.

[35] Justin Martineau and Tim Finin. Delta TFIDF: An improved feature space for sentiment analysis. In *ICWSM*, 2009.

[36] Rada Mihalcea and Carlo Strapparava. Learning to laugh (automatically): Computational models for humor recognition. *Computational Intelligence*, 22(2):126–142, 2006.

[37] Tony Mullen and Nigel Collier. Sentiment analysis using support vector machines with diverse information sources. In *EMNLP*, volume 4, pages 412–418, 2004.

[38] Jin-Cheon Na, Haiyang Sui, Christopher Khoo, Syin Chan, and Yunyun Zhou. Effectiveness of simple linguistic processing in automatic sentiment classification of product reviews. *Advances in Knowledge Organization*, 9:49–54, 2004.

[39] C. Manning P. Beineke, T. Hastie and S. Vaithyanathan. Exploring sentiment summarization. In *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text, AAAI technical report*, pages 04–07, 2004.

[40] Alexander Pak and Patrick Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of LREC*, volume 2010, 2010.

[41] Georgios Paltoglou, Stephane Gobron, Marcin Skowron, Mike Thelwall, and Daniel Thalmann. Sentiment analysis of informal textual communication in cyberspace, 2010.

[42] Bo Pang and Lillian Lee. A sentimental education : Sentiment analysis using subjectivity summarization based on minimum cuts. *Cornell University Library*, 2004.

[43] Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135, January 2008.

[44] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of EMNLP*, pages 79–86, 2002.

[45] Soujanya Poria, Alexander F. Gelbukh, Erik Cambria, Dipankar Das, and Sivaji Bandyopadhyay. Enriching senticnet polarity scores through semi-supervised fuzzy clustering. In Jilles Vreeken, Charles Ling, Mohammed Javeed Zaki, Arno Siebes, Jeffrey Xu Yu, Bart Goethals, Geoffrey I. Webb, and Xindong Wu, editors, *ICDM Workshops*, pages 709–716. IEEE Computer Society, 2012.

[46] C. Banea R. Mihalcea and J. Wiebe. Learning multilingual subjective language via cross-lingual projections. In *Proceedings of the Association for Computational Linguistics (ACL)*, pages 976–983, 2007.

[47] Ellen Riloff and Janyce Wiebe. Learning extraction patterns for subjective expressions. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 105–112, Morristown, NJ, USA, 2003. Association for Computational Linguistics.

[48] Ellen Riloff, Janyce Wiebe, and William Phillips. Exploiting subjectivity classification to improve information extraction. In *Proceedings of the National Conference On Artificial Intelligence*, volume 20, page 1106. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2005.

[49] Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[50] Pero Subasic and Alison Huettner. Affect analysis of text using fuzzy semantic typing. *Fuzzy Systems, IEEE Transactions on*, 9(4):483–496, 2001.

[51] J. Wiebe T. Wilson and R. Hwa. Just how mad are you? finding strong and weak opinion clauses. In *Proceedings of AAAI*, volume 22, page 761769, 2006.

[52] Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly D. Voll, and Manfred Stede. Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2):267–307, 2011.

[53] Hiroya Takamura, Takashi Inui, and Manabu Okumura. Latent variable models for semantic orientations of phrases. In *Proceedings of the 11th Meeting of the European Chapter of the Association for Computational Linguistics (EACL 2006)*, 2006.

[54] Peter Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. pages 417–424, 2002.

[55] Hongning Wang, Yue Lu, and Chengxiang Zhai. Latent aspect rating analysis on review text data: a rating regression approach. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 783–792. ACM, 2010.

[56] Casey Whitelaw, Navendu Garg, and Shlomo Argamon. Using appraisal groups for sentiment analysis. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 625–631. ACM, 2005.

[57] J. Wiebe and R. Mihalcea. Word sense and subjectivity. In *Proceedings of the Conference on Computational Linguistics / Association for Computational Linguistics (COLING/ACL)*, 2006.

[58] J. Wiebe and T. Wilson. Learning to disambiguate potentially subjective expressions. In *Proceedings of the Conference on Natural Language Learning (CoNLL)*, pages 112–118, 2002.

[59] Janyce M. Wiebe. Learning subjective adjectives from corpora. In *In AAAI*, pages 735–740, 2000.

[60] Yorick Wilks and Mark Stevenson. The grammar of sense: Using part-of-speech tags as a first step in semantic disambiguation. *Natural Language Engineering*, 4(2):135–143, 1998.

[61] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 347–354. Association for Computational Linguistics, 2005.

[62] Ian H Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.

[63] Kiduk Yang, Ning Yu, Alejandro Valerio, and Hui Zhang. Widit in trec 2006 blog track. In *TREC*, 2006.

[64] Hong Yu. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. *Proceedings of the 2003 conference on Empirical methods in Natural Language Processing*, 2003.

[65] Hong Yu. Towards answering opinion questions?: Separating facts from opinions and identifying the polarity of opinion sentences. *Proceeding EMNLP ?03 Proceedings of the 2003 conference on Empirical methods in natural language processing*, 2003.

[66] Zhongwu Zhai, Bing Liu, Hua Xu, and Peifa Jia. Grouping product features using semi-supervised learning with soft-constraints. In Chu-Ren Huang and Dan Jurafsky, editors, *COLING 2010, 23rd International Conference on Computational Linguistics, Proceedings of the Conference, 23-27 August 2010, Beijing, China*, pages 1272–1280. Tsinghua University Press, 2010.

[67] Ethan Zhang and Yi Zhang. Ucsc on rec 2006 blog opinion mining. In *TREC*, 2006.

[68] Jun Zhao, Kang Liu, and Gen Wang. Adding redundant features for crfs-based sentence sentiment classification. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 117–126, 2008.