Sentiment Analysis Using Domain-Adaptation and Sentence-Based Analysis

Gizem Gezici, Berrin Yanikoglu, Dilek Tapucu and Yücel Saygın

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 apply a simple method to adapt a general purpose polarity lexicon to a specific domain [1]. Subsequently, we propose and evaluate new features to be used in a word polarity based approach to sentiment classification. In particular, we analyze sentences as the first step for estimating the overall review polarity. We consider different aspects of sentences, such as length, purity, irrealis content, subjectivity, and position within the opinionated text. This analysis is then used to find sentences that may convey better information about the overall review polarity. We use a subset of hotel reviews from the TripAdvisor database [2] to evaluate the effect of sentence-level features on sentiment classification. Then, we measure the performance of our sentiment analysis engine using the domain-adapted lexicon on a large subset of the TripAdvisor database.

G. Gezici · B. Yanikoglu (✉) · Y. Saygın
Faculty of Engineering and Natural Sciences, Sabanci University, 34956 Istanbul, Turkey
e-mail: berrin@sabanciuniv.edu

G. Gezici
e-mail: gizemgezici@sabanciuniv.edu

Y. Saygın
e-mail: ysaygin@sabanciuniv.edu

D. Tapucu
Department of Computer Engineering, Izmir Institute of Technology, 35430 Izmir, Turkey
e-mail: dilektapucu@sabanciuniv.edu

M.M. Gaber et al. (eds.), *Advances in Social Media Analysis*,
Studies in Computational Intelligence 602,
DOI 10.1007/978-3-319-18458-6_3

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 *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 [3]. In the remainder of this chapter, we use the terms "document", "review" and "text" interchangeably, to refer to the text whose sentiment polarity or opinion strength is to be estimated.

Two main approaches for sentiment analysis are defined in the literature: one approach is called *lexicon-based* [4] and the other is based on *supervised learning* [5]. The lexicon-based approach calculates the semantic orientation of a given text from the polarities of the constituent words or phrases [4], obtained from a lexicon such as the SentiWordNet [6]. In this approach, different features of the text may be extracted from word polarities [7], such as average word polarity, or the number of subjective words, but the distinguishing aspect is that there is no supervised learning. Furthermore, the text is often treated as a *bag-of-words*; in other words, features are obtained from constituent words without keeping track of the location of those words. Alternatives to the bag-of-word approach are also possible, where word polarities of the first sentence etc. are calculated separately [8]. 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 [6] and SenticNet [9] are two widely known domain-independent lexicons.

Supervised learning approaches use machine learning techniques to establish a model from an available corpus of reviews with associated labels. For instance in [5, 10], researchers use the Naive Bayes algorithm to separate positive reviews from negative ones by learning the conditional 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 Latent Dirichlet Analysis (LDA) approach, a training corpus is used to learn the probability distributions of topic and word occurrences 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 [11, 12]. While supervised

approaches are typically more successful than lexicon-based ones, collecting a large amount of labelled, domain-specific data can be a problem.

In this work, we present a supervised learning approach to sentiment analysis, addressing two sub-tasks of the problem. First, we apply a simple domain-adaptation method proposed in [1] to adapt a domain-independent polarity lexicon to a specific domain. We show that even changes in the polarity of a small number of words affect the overall accuracy by a few percent. Next, we propose a sentence-based analysis of the review sentiment, using the updated polarity lexicon for 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. The use of sentence-based analysis is aimed to bridge this gap.

The remainder of this chapter is organized as follows. Section 2 provides an overview of related work. Section 3 proposes the adaptation process of a domain-independent lexicon. Section 4 describes our sentence-based sentiment analysis approach that forms the main contribution in this work. Section 5 presents the learning module and Sect. 6 reports experimental results. Finally, in Sect. 7 we conclude and outline our ideas for future work.

2 Related Work

We summarize related work in three sections: we describe some of the important work in sentiment analysis to give the general overview; followed by work on adaptation of a domain-independent polarity lexicon; and finally work that use a sentence-based or similar approach.

Research in sentiment analysis has started in the last 10–12 years, with increasing academic and commercial interest to the field. An elaborate survey of the previous works for sentiment analysis has been presented in [3] while we only summarize some important trends here.

In the earlier works, the document is typically viewed as a *bag of words* and its sentiment polarity is estimated from the average polarity of the words inside the document [5, 13–15]. Since looking at the whole document only as a bag of words is very simplistic, later work focused on analysis of phrases and sentences. Among these, some focused on subjectivity analysis of phrases/sentences, so as to make use of this information while determining the subjectivity of the document. In one of the early studies, Wiebe discovered subjective adjectives from corpora [16]. Then, Hatzivassiloglou and Wiebe [16] 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. Subsequently, several studies focused on sentence-level or sub-sentence-level subjectivity detection in different domains. Some recent work also examined relations between word sense disambiguation and subjectivity, in order to extract sufficient information for a more accurate sentiment classification [18]. Wiebe et al.

introduces a broad survey of subjectivity recognition using various features and clues [19].

Determining the sentiment strength or polarity value of a given document, rather than simply classifying it as positive or negative, is a *regression* problem that is adressed using slightly different supervised learning techniques. In a regression problem, the task is to learn the mapping $y = f(x)$, where $x \in R^d$ and $y \in R$. It can be said that the regression problem is more difficult than the classification problem, as the latter can be accomplished once sentiment polarities are estimated. If one considers three sentiment categories (negative, neutral and positive), then treating the problem as a regression problem rather than a classification problem, may be the more appropriate approach since class labels are ordinal.

As the number of classes increase, the classification task becomes more difficult. For instance, classifying a review as positive or negative (two-class classification) is much easier than classifying it as very negative, negative, neutral, positive, and very positive (five-class problem). The problem is even more difficult when one considers objective as a separate category (e.g. negative, neutral, positive and objective), since objective and other (often neutral) classes may carry similar sentiment values. Here, the reader should note that opinionated text can be neutral ("The hotel was so so") and objective text can carry a sentiment value ("The hotel lacks a pool"). In approaching this problem, determining the sentiment subjectivity and sentiment strength can be done in two-steps.

A *polarity lexicon* indicates the sentiment polarity of words or phrases. SentiWordnet [4] and SenticNet [9] are two of the most commonly used polarity lexicons, for sentiment analysis. In [20], authors discuss three main approaches for opinion lexicon building: manual approach, dictionary-based approach, and corpus-based approach. The major shortcoming of the manual approach (e.g. [21]) is the cost (time and effort) to hand select words to build such a lexicon. There is also the possibility of missing important words that could be captured with automatic methods. Dictionary-based approaches (e.g. [4, 13, 22]) work by expanding a small set of seed opinion words, with the use of a lexical resource such as the WordNet [23]. Note that with these approaches, the resulting lexicons are domain-independent.

Corpus-based approaches can be used to learn domain-specific lexicons using a domain corpus of labeled reviews. Wilson et al. stress the importance of contextual polarity to differentiate from the prior polarity of a word [24]. They extract contextual polarities by defining several contextual features. In [25], a double propagation method is used to extract both sentiment words and features, combined with a polarity assignment method starting with a seed set of words. In [26], authors use linear programming to update the polarity of words based on specified hard or soft constraints. Another application of linear programming appears in [27] to learn a sentiment lexicon which is not only domain specific but also aspect-dependent. Another recent work expands a given dictionary of words with known polarities by first producing a new set of synonyms with polarities and using these to further deduce the polarities of other words [28]. Finally, a simple corpus-based domain adaptation technique proposed by Demiroz et al. is used in our system [1]. In this work, authors consider

the tf-idf [29] scores of each word in positive and negative review sets, and adapt word polarities according to this difference.

The idea of sentence-level analysis is not new. Some researchers approached the problem by first finding subjective sentences in a review, with the hope of eliminating irrelevant sentences that would generate noise in terms of polarity estimation [8, 30]. Yet another approach is to exploit the structure in sentences, rather than seeing a review as a bag of words [13–15]. For instance in [13], conjunctions were analyzed to obtain the polarities of the words that are connected with the conjunct. In addition, Wilson et al. [32] raise the question of obtaining clause-level opinion strength as a preparation step for sentence-level sentiment analysis. In [33, 34] researchers focused on sentence polarities separately, again to obtain sentence polarities more correctly, with the goal of improving review polarity in turn. The first line polarity has also been used as a feature by [8].

Our work is motivated by our observation that the first and last lines of a review are often very indicative of the review polarity. Starting from this simple observation, we formulated more sophisticated features for sentence level sentiment analysis. In order to do that, we performed an in-depth analysis of different sentence types.

Our approach described in the remainining sections has two main parts: domain-adaptation of a general purpose polarity lexicon and sentiment analysis using the adapted lexicon and new, sentence-based features. We explain these two parts in Sects. 3.2 and 4, 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 [1], by using a simple sentiment analysis algorithm with and without domain adaptation of the used lexicon. We use the adapted polarity lexicon for feature extraction. For evaluating the sentiment of a given text, 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 Sect. 6 show the effectiveness of the proposed method.

3 Domain-Adaptation of a Polarity Lexicon

3.1 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 $\langle pol^-, pol^=, pol^+ \rangle$ for each word [6]. The polarity scores indicate the measure of negativity, objectivity and positivity, and they sum up to 1. Some sample scores are provided in Table 1 from SentiWordNet.

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:

Table 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

$$pol(w) = \begin{cases} 0 & \text{if } \; max(pol^{=}, pol^{+}, pol^{-}) = pol^{=} \\ pol^{+} & \text{else if } \; pol^{+} \geq pol^{-} \\ -pol^{-} & \text{otherwise} \end{cases} \tag{1}$$

In other words, given the polarity triplet $\langle pol^{-}, pol^{=}, pol^{+} \rangle$ 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 $\langle 0.75, 0.125, 0.125 \rangle$ $pol(\text{"sufficient"}) = -0.75$. Similarly, the polarity triplet of the word "moldy" is $\langle 0.375, 0.625, 0.0 \rangle$; hence $pol(\text{"moldy"}) = 0$.

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 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. For domain adaptation, we use the technique proposed in [1] with their best reported update mechanism. The proposed approach allows 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. It was shown

in [1] that updating the polarities of even a small set of words has a significant contribution to sentiment analysis accuracy.

This method analyzes the occurrence of the words in the lexicon in positive and negative reviews in a given domain. If a particular word occurs significantly more often in positive reviews than in negative reviews, then it is assumed that this word should have positive polarity for this domain, and vice versa. In this case, the polarity of that word is updated in the domain-specific lexicon.

While any domain-independent polarity lexicon can be used, we have adapted a commonly used lexicon, namely SentiWordNet [6]. Results with bigger and better lexicons such as SenticNet [9] are expected to be similar, albeit possibly showing smaller benefits.

In this method, inspired by [35], the tf-idf (term frequency-inverse document frequency) scores of each word is calculated 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') [36]. There are quite a few variants of tf-idf computations [29], and the tf-idf variant used by Demiroz et al. [1] is computed as:

$$\begin{aligned} tf.idf(w_i, +) &= tf(w_i, +) \times idf(w_i) = log_e(tf(w_i, +) + 1) \times log_e(N/df(w_i)) \\ tf.idf(w_i, -) &= tf(w_i, -) \times idf(w_i) = log_e(tf(w_i, -) + 1) \times log_e(N/df(w_i)) \end{aligned} \tag{2}$$

where the first term to the right of the equality 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. Then, the term $(\Delta tf)idf$ is defined as:

$$(\Delta tf)idf(w_i) = tf.idf(w_i, +) - tf.idf(w_i, -) \tag{3}$$

Demiroz et al. [1] considers different alternatives about which polarities to update (e.g. the ones for which the $(\Delta tf)idf$ magnitude is large) and how to update them (e.g. use the $(\Delta tf)idf$ value as polarity or shift the original polarity value) and show that updating even a small percentage of all words in a lexicon improves sentiment analysis.

For adapting the domain-specific lexicon, we use the same update algorithm along with the best update method found in this work [1]. In this update method, we shift the polarities of the words that have the largest $(\Delta tf)idf$ scores in terms of absolute values. Hence, we consider both the largest and smallest $(\Delta tf)idf$ scores, suggesting the words with positive and negative connotations respectively. Once we select which words to adapt, we shift the original polarity values of those words towards their $(\Delta tf)idf$ scores by 0.4.

Table 2 Summary of features

Group name	Feature	Name
Basic	F_1	Average review polarity
	F_2	Review purity
	F_3	Review subjectivity
$(\Delta tf)idf$	F_4	Total $(\Delta tf)idf$ scores of all words
	F_5	Average review polarity, weighted by $(\Delta tf)idf$ scores
Seed words statistics	F_6	Freq. of seed words
	F_7	Avg. polarity of seed words
	F_8	Stdev. of polarities of seed 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}	Total $(\Delta tf)idf$ scores of words in the first line
	F_{18}	$(\Delta tf)idf$ weighted polarity of first line
	F_{19}	Total $(\Delta tf)idf$ scores of words in the last line
	F_{20}	$(\Delta tf)idf$ weighted polarity of last line
	F_{21}	Number of sentences in review
	F_{22}	Avg. pol. of subj. sentences
	F_{23}	Avg. pol. of pure sentences
	F_{24}	Avg. pol. of non-irrealis sentences

4 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. The 24 features can be grouped in five listed in Table 2: (1) basic features, (2) $(\Delta tf)idf$ weighting of word polarities, (3) features based on the seed words statistics, (4) punctuation, and (5) sentence-level features.

Our approach depends on the existence of a sentiment lexicon that provides information about the semantic orientation of single or multiple terms. Specifically, we use the SentiWordNet [6] as the base lexicon and its domain-adapted version for domain-specific lexicon.

In the following sections, we define a review R as a sequence of sentences $R = S_1 S_2 S_3, \ldots, S_M$ where M is the number of sentences in R. The review R is also viewed as a sequence of words $w_1, \ldots, w_T$, where T is the total number of words in the review.

4.1 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. We only include words with POS tags containing "NN", "JJ", "RB", and "VB" since these are the words that are possible sentiment-baring words in a review.

$$\text{Average review polarity} = \frac{1}{T} \sum_{j=1..T} pol(w_j) \tag{4}$$

$$\text{Review purity}(\sum_{j=1..T} pol(w_j))/(\sum_{j=1..T} |pol(w_j)|) \tag{5}$$

The review subjectivity is a binary variable that is 1 if one of the sentences in the review is deemed as subjective, as defined in Sect. 4.

4.1.1 Δtf*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 as explained in Sect. 3.2. 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.

We then extract two features using the $(\Delta tf)idf$ scores. In feature F_4, we compute the sum of the $(\Delta tf)idf$ scores of the unique words in a review. We expect that this feature may replace or complement the average review polarity obtained from the domain-independent lexicon. Note that the average $(\Delta tf)idf$ score of the words in the review would be very similar to the average polarity of the words (F_1) in the review; hence we preferred to use the sum even though it is dependent on the review length. As another feature F_5, we tried combining the two sources of information, where we weighted the polarities of all words in the review by their $(\Delta tf)idf$ scores (F_5).

Table 3 Chosen seed words

Positive word	Type	Negative word	Type
Great	JJ	Room	NN
Excellent	JJ	Desk	NN
Wonderful	JJ	Never	RB
Perfect	JJ	Worst	JJS
Fantastic	JJ	Manager	NN
Comfortable	JJ	Bad	JJ
Helpful	JJ	Night	NN
Friendly	JJ	Even	RB
Location	NN	Terrible	JJ
Lovely	JJ	Rude	JJ

4.1.2 Seed Word Statistics

Like some other researchers, we also use a smaller subset of the lexicon consisting of 20 clearly positive and 20 clearly negative seed words for the given domain, with the hope that they may indicate the reviews polarity with more certainty. To appreciate this approach, one can note that while a negative sentence might contain the word "good" ("the food was not good"), it is less likely for a negative sentence to contain the word "excellent" (e.g. "the food was not excellent"). In general, it is more likely to see a negative sentence containing a positive term, than a negative sentence containing a clearly positive seed word.

While we have computed the seed words automatically by analyzing the $(\Delta tf)idf$ scores of words, we assume that forming such a small list manually is feasible for any domain. To determine the seed words in the given domain, we first compute the $(\Delta tf)idf$ scores of all unique words in the corpus. Then, we sort these words by using their $(\Delta tf)idf$ scores and selected the top-20 positive and top-20 negative words in the list. These words then form the seed word set, called *SeedW*. We include a sub-sample of 10 positive and 10 negative seed words in Table 3.

We then define $SeedW(R)$ as the set of seed words that appear in review R and extract three features related to seed words in the review (features $F6 - F8$):

$$\text{Freq. of seed words} = |SeedW(R)|/|R| \tag{6}$$

$$\text{Avg. polarity of seed words} = \frac{1}{|SeedW(R)|} \sum_{w_j \in SeedW(R)} pol(w_j) \tag{7}$$

We also include the standard deviation of the polarity of seed words in the review, to capture if there are any disagreements.

4.1.3 Punctuation Features

We have four features related to punctuation. Two of these features were suggested before; namely, the number of exclamation marks and the number of question marks [37]. Note that an exclamation mark typically makes the stated emotion stronger ("the food was good!!"), but it can also be used to indicate an incredulous reaction (e.g. "the room did not have a window!"). On the other hand, the question mark can be used to detect objective/neutral sentences that may be otherwise classified as having sentimental polarity ("are the rooms big?").

Emoticons, in our case positive and negative smiley-faces, are also important sentiment-bearing symbols, as proposed in [38, 39]. As with the exclamation and question marks, smiley-faces may also have distinct context-specific meanings. For instance the positive smiley may be used positively, to indicate happiness (e.g. "the room had a view :)") or to make fun of something or agree with a joke.

Despite the ambiguities, we have included the two punctuation features and the two smiley-faces in our feature set, with the hope that statistics related to their usage may give some additional information to the classifier.

4.1.4 Sentence-Level Features

Often the first or last line in a review summarizes the overall review sentiment. This is certainly true for long reviews found in hotel reviews. For instance a title or first line such as "Excellent hotel!" clearly denotes the overall sentiment, no matter what is said in the details of the review. In this work, we propose to consider the review as a set of sentences and estimate the review sentiment by considering the types and sentiment strength of the constituent sentences.

Sentence-level features are extracted from (i) sentences in certain locations in the review (e.g. the first and last lines of the review) or (ii) certain types of sentences (e.g. subjective sentences). In particular, we consider subjective, pure and non-irrealis sentences and use features extracted from such sentences for detecting the review sentiment.

There are many possibilities in a sentence-based analysis. For instance, one can (i) consider only subjective sentences or (ii) use the features of subjective sentences as additional features in the system. We explored both of these approaches and then decided to add sentence-level features to the system.

In order to identify subjective sentences, we looked at if a sentence contains at least one subjective word or a smiley; if so that sentence is deemed as subjective. For subjectivity of the word, we adopted the same idea that was proposed in [40].

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. 5, using only the words in the sentence. We experimented with different values of τ and for evaluation we used $\tau = 0.8$.

Table 4 Sentence-level features for a review R

F_{13}	Avg. first line polarity	$\frac{1}{\lvert S_1\rvert}\sum_{w\in S_1} pol(w)$
F_{14}	Avg. last line polarity	$\frac{1}{\lvert S_M\rvert}\sum_{w\in S_M} pol(w)$
F_{15}	First line purity	$[\sum_{w\in S_1} pol(w)]/[\sum_{w\in S_1} \lvert pol(w)\rvert]$
F_{16}	Last line purity	$[\sum_{w\in S_M} pol(w)]/[\sum_{w\in S_M} \lvert pol(w)\rvert]$
F_{17}	$(\Delta tf)idf$ weighted polarity of 1st line	$(\sum_{w\in S_1} \Delta tf * idf(w)) \times pol(w)$
F_{18}	Total $(\Delta tf)idf$ scores of 1st line	$\sum_{w\in S_1} \Delta tf * idf(w)$
F_{19}	$(\Delta tf)idf$ weighted polarity of last line	$(\sum_{w\in S_M} \Delta tf * idf(w)) \times pol(w)$
F_{20}	Total $(\Delta tf)idf$ scores of last line	$\sum_{w\in S_M} \Delta tf * idf(w)$
F_{21}	Number of sentences in review	M
F_{22}	Avg. pol. of subj. sentences	$\frac{1}{\lvert subjS(R)\rvert}\sum_{w\in subjW(R)} pol(w)$
F_{23}	Avg. pol. of pure sentences	$\frac{1}{\lvert pureS(R)\rvert}\sum_{w\in pure(R)} pol(w)$
F_{24}	Avg. pol. of non-irrealis sentences	$\frac{1}{\lvert nonIrS(R)\rvert}\sum_{w\in nonIr(R)} pol(w)$

We also looked at sentences containing *irrealis* terms, as they indicate the opposite sentiment than the sentiment carried by the constituent words (e.g. "I thought the hotel would have nicer rooms."). In order to determine irrealis sentences, the existence of the modal verbs "would", "could", or "should" is checked. If one of these modal verbs appears in the sentence, then these sentences are labeled as irrealis sentences, as was the case in [7]. Then, we chose *non-irrealis* sentences as our third sentence type for analysis.

These three sets of sentences in a review R are called $subS(R)$, $pureS(R)$ and $nonIrS(R)$. The sentence-based features ($F_{13} - F_{24}$) are given in Table 4:

We tried three different approaches for this purpose.

- In the first approach, each review is pruned to keep only the sentences that are possibly more useful (e.g. only subjective sentences) for sentiment analysis. For pruning, thresholds were set separately for each sentence-level feature. Sentences with absolute purity of at least 0.8 are defined as pure sentences. Pruning sentences in this way resulted in lower accuracy in general, due to loss of information.
- In the second approach, the polarities in special sentences (pure, subjective or no irrealis) are given higher weights while computing the average review polarity. In effect, this is a soft version of pruning, as the other sentences are given lower weight, rather than a weight of zero.
- In the third approach, we used the information extracted from sentence-level analysis as additional features (e.g. average polarity of subjective sentences were added as a new feature). This approach gave the best results and is used in the final system.

5 Classification

Given a review, we first apply feature extraction and represent the review by its features given in Table 2. We used a supervised learning approach to train a classifier to learn to classify a review into different sentiment classes or to assign a sentiment strength to it.

For two-class classification, we took 1- and 2-star reviews in the training set as negative samples and 4- and 5-star reviews as positive samples. For three-class classification, we instead trained a regression engine to estimate the sentiment strength and then decided on two thresholds delineating the negative-neutral and neutral-positive boundaries.

In the classification problem (2- to 5-class classification problems are considered for the hotel domain in literature), the performance measure is the classification accuracy; in other words, what percentage of queried reviews got classified correctly. In the case of regression, a natural error measure is the Mean Squared Error (MSE) or Mean Absolute Error (MAE), in other words the average squared or absolute error between the estimated strength and the ground truth. When using regression as a first step for classification, one can measure classification accuracy by using thresholds after estimating the sentiment strength.

As classifier, we used the Support Vector Machines (SVM) [41]. For the implementation, we used the LibSVM [43]. package in WEKA [44] for both train and test phases. The SVM requires two main parameters for training: the kernel and the cost (C) parameter. The kernel, cost and gamma parameters required for one of the kernels were decided on the validation set, using WEKA. For kernel, we tried the RBF & linear kernels and observed that the RBF kernel worked better than the linear kernel for our task.

For two-class classification, we used C-SVC (classification), RBF kernel and the parameter pair (10.0, 10.0). For regression, we used epsilon-SVR (regression) as SVM type and set the normalization to true by default. The cost and gamma parameters were the same as for classification, even though parameter optimization was done separately for this problem.

6 Experimental Evaluation

In this section, we provide an evaluation of the sentiment analysis engine. Our evaluation procedure is composed of two main parts. First, we report the effectiveness of different sets of features using star-rated reviews from the TripAdvisor website [2]. Next, we evaluate our overall system with state-of-the-art approaches on the hotel reviews dataset presented in [12].

6.1 Dataset

The TripAdvisor dataset consists of around 240,000 customer-supplied reviews of 1,850 hotels and was introduced by [42]. 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/her evaluation.

For feature and overall system evaluation, we used a publicly available dataset that was collected from this corpus [43], in order to make the system comparable to the state-of-the-art approaches. This dataset contains around 90,000 hotel reviews, in three subsets: the train, validation and test subsets contain approximately 76,000, 6,000 and 13,000 reviews respectively. Each of these three subsets contains a balanced number of negative (1-star and 2-star) and positive (4-star and 5-star) reviews.

The dataset also includes neutral reviews (e.g. with a rating value of 3) that are used in three-class classification. For binary classification, these neutral reviews are omitted from the dataset.

For feature evaluation task, we first used the validation subset to select the best feature subsets using two appropriate classifiers on WEKA [44]. The validation dataset is also used to find the best parameters for the corresponding classifiers. Then, the test dataset is used to evaluate feature subsets and overall performance of the system, with the two selected classifiers.

6.2 Implementation

We computed $\Delta tf * idf$ scores of the words which have POSTags of noun, adjective, verb and adverb in the training set. Subsequently, we updated the dominant polarities of the words obtained from SentiWordNet [45] according to the polarity adaptation procedure explained in Sect. 3.2.

We calculated features as explained in Sect. 4 and generated intermediate files that represent a review as a set of features, along with its label. These intermediate files for the three sub-datasets (e.g. train, validation and test) were created by a Java implementation on Eclipse environment and given to WEKA [44].

For classification, we trained a Support Vector Machine classifier with the Sequential Minimal Optimization (SMO) algorithm, using the intermediate files as training data. For this, we used the LibSVM library which is included in WEKA environment [44]. Firstly, we observed that the RBF kernel worked better than other kernels for our purpose. Then, we found the best parameter pair for the cost and gamma parameters of this kernel and evaluated our overall system with these optimal paramters on WEKA [44].

For the purpose of various feature subsets evaluation, we used two different classifiers (SMO and Logistic) that are also integrated into WEKA environment [44] after we tried several other classifiers.

Table 5 The effect of feature subsets on two-class classification using the TripAdvisor Dataset [12]

Feature Subset	Accuracy (SMO) (%)	Accuracy (Logistic) (%)
Basic(F1-F3)	59.62	59.66
... + $(\Delta tf)idf$(F4-F5)	59.97	59.48
... + Seed Words(F6-F8)	59.97	59.48
... + Punctuation(F9-F12)	60.47	60.18
... + First&LastLine Avg. pol. and Purity(F13-F16)	60.60	60.62
... + First&LastLine $(\Delta tf)idf$(F17-F20)	60.74	60.67
... + Sentence count(F21)	60.70	60.78
... + Subj. Sentence Avg. pol.(F22)	63.76	64.27
... + Pure Sentence Avg. pol.(F23)	63.21	62.89
... + Non-Irrealis Sentence Avg. pol.(F24)	**63.76**	**64.27**
Basic + Seed Words(F6-F8)	59.62	59.66
Basic + Punctuation(F9-F12)	60.11	60.03
Basic + First&LastLine Avg. pol. and Purity(F13-F16)	59.97	59.94
Basic + First&LastLine $(\Delta tf)idf$(F17-F20)	60.28	59.72
Basic + Sentence count(F21)	60.05	59.93
Basic + Subj. Sentence Avg. pol.(F22)	61.27	60.27
Basic + Pure Sentence Avg. pol.(F23)	60.19	60.02
Basic + Non-Irrealis Sentence Avg. pol.(F24)	62.47	62.64

6.3 Results

6.3.1 Contributions of Feature Subsets on Overall Accuracy

The accuracies obtained on the two-class problem are given in Table 5 where there are two groups of results. In the upper half of the table, we provide the results as more features are incrementally added in the order listed in Table 2. Note that in this way, features that are added first have more of a chance to improve on the baseline accuracy, so in the lower half, we provide accuracy results when features are added one by one to the basic features.

When considering these results, we noted that most of the accuracy gains were obtained with punctuation features (59.97 to 60.47 % using SMO); the addition of subjective sentences (60.70 to 63.76 % using SMO); and the addition of non-irrealis features (63.21 to 63.76 % using SMO).

Also, we noted that there is no improvement when features related to seed words statistics are added on top of the basic plus $(\Delta tf)idf$ features. This shows that seed words related features do not bring extra information to the system. Although, seed words seem to have no effect on overall accuracy, we still included seed words statistics features for the sake of completeness.

6.3.2 Overall Engine Comparison with Previous Systems

We provide state-of-the-art sentiment analysis performance results obtained using reviews from the TripAdvisor website in Table 6. Unfortunately not all systems report results that are directly comparable: they may differ in the tested data set; the reported accuracy or error measure; or the classification problem. In Table 6, different systems are grouped according to the number of classes. For instance some systems have reported their performance in the binary classification problem of separating 1- or 2-star reviews, from the 4- or 5-star reviews.

As one can see, the best results so far are obtained by Bespalov et al. using the LDA approach, with 6.90 % error rate on the binary classification problem [11, 43], while our error rate for this task is 13.23 %. In terms of the f-measure, our results surpass previously reported f-measures, with an f-measure of 0.87 for the binary classification problem and 0.64 for the three-class problem.

For the 5-class classification task, the best results achieved so far are again by Bespalov et al. with 40.76 % error rate [11, 43]. We handled the 5-class classification task as a regression problem and obtained the regression values for class labels of reviews from WEKA [44]. This gave a Mean Absolute Error of 0.43 on the test set. This can be seen as a review with +1 target value being assigned a sentiment strength of 0.57 (1 − 0.43). When we rounded the estimated regression values (e.g. 1.8 became 2 while 1.3 became 1) and obtained classification in this way, the misclassification error is measured to be 56.25 %. While this rate is high, it actually highlights the difficulty of the 5-class classification problem. Note that a random classifier would be expected to be accurate for about one in five cases, or have an error rate of 80 %.

6.3.3 Discussion

As can be seen in Table 6, our system with the newly proposed features provides one of the best results obtained so far, except for the work presented in [11, 12].

It is noteworthy to mention that [12] is a more recent version of [11] and they both use LDA as the core approach. 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.

Table 6 State of the art results on the TripAdvisor Corpus

Previous work	Dataset	F-measure	Error rate	Task
Peter et al. [46]	103000	0.83	–	Binary: 1 versus {4,5}
Gindl et al. [47]	1800	0.79	–	Binary: {1,2} versus {4,5}
Gezici et al. [48]	6000	0.81	–	Binary: {1,2} versus {4,5}
Bespalov et al. [11]	96000[a]	–	7.37	Binary: {1,2} versus {4,5}
Bespalov et al. [12]	96000	–	6.90	Binary: {1,2} versus {4,5}
This work	96000	0.87	13.23	Binary: {1,2} versus {4,5}
Grabner et al. [49]	1000	0.55	–	Three-class: {1,2}, {3}, {4,5}
This work	96000	0.64	36.50	Three-class: {1,2}, {3}, {4,5}
Bespalov et al. [11]	96000[a]	–	49.20	Five-class
Bespalov et al. [12]	96000	–	40.76	Five-class
This work	96000	–	56.25	Five-class

[a]This dataset is different than the dataset released by Bespalov et al. [12]; so the results are not directly comparable

7 Conclusions and Future Work

We tried to bridge the gap between word-level polarities and review-level polarity through an intermediate step of sentence-level analysis of the reviews. We formulated new features for sentence-level sentiment analysis by an in-depth analysis of the sentences.

We implemented the proposed features and evaluated them on a publicly available dataset of TripAdvisor reviews [12], to show the effect of sentence-level features on polarity classification. We observed that sentence-level features indeed have an effect on sentiment classification accuracy; therefore, we conclude that sentences do matter in sentiment analysis and they may be even more useful in more diverse datasets such as blogs.

We also evaluated our domain-adapted engine on the same dataset of TripAdvisor hotel reviews and summarized state-of-the-art results in that domain. The variability of the datasets and accuracy measures make the reported results difficult to compare directly. Nonetheless, one can observe that two-class classification of text into positive and negative classes can be done quite robustly, while the five-class classification (required for assigning a star-rating) requires more work.

As future work, we will consider using word embeddings that have been shown to be successful in different problems [47], along with our existing approach. Sentence-based analysis can also be explored further to identify essential sentences in a review or for highlighting important sentences for review summarization.

Acknowledgments This work was partially funded by European Commission, FP7, under UBIPOL (Ubiquitous Participation Platform for Policy Making) Project (www.ubipol.eu). Dr. Dilek Tapucu was a post-doctoral researcher at Sabanci University at the time of this project.

References

1. Demiroz, G., Yanikoglu, B. Tapucu, D., Saygin, Y.: Learning domain-specific polarity lexicons, In: 2012 IEEE 12th International Conference on Data Mining Workshops (ICDMW), pp. 674–679 (2012)
2. The TripAdvisor website. http://www.tripadvisor.com [TripAdvisor LLC]. Accessed in 2012
3. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Found. Trends Inf. Retrieval **2**(1–2), 1–135 (2008)
4. Turney, P.D.: Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, pp. 417–424. Association for Computational Linguistics (2002)
5. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 conference on Empirical methods in natural language processing, vol. 10, pp. 79–86. Association for Computational Linguistics (2002)
6. Esuli, A., Sebastiani, F.: SentiWordNet: a publicly available lexical resource for opinion mining. In: Proceedings of the 5th Conference on Language Resources and Evaluation (LREC06), pp. 417–422 (2006)
7. Taboada, M., Brooke, J., Tofiloski, M., Voll, K.D., Stede, M.: Lexicon-based methods for sentiment analysis. Comput. Linguist. **37**(2), 267–307 (2011)
8. Zhao, J., Liu, K., Wang, G.: Adding redundant features for crfs-based sentence sentiment classification. In: Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pp. 117–126 (2008)
9. Poria, S., Gelbukh, A.F., Cambria, E., Das, D., Bandyopadhyay, S.: Enriching SenticNet polarity scores through semi-supervised fuzzy clustering. In: Vreeken, J., Ling, C., Zaki, M.J., Siebes, A., Yu, J.X., Goethals, B., Webb, G.I., Wu, X. (eds.) ICDM Workshops, pp. 709–716. IEEE Computer Society (2012)
10. Yu, H., Hatzivassiloglou, V.: Towards answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences. In: Proceedings of the 2003 conference on Empirical methods in Natural Language Processing, pp. 129–136. Association for Computational Linguistics (2003)
11. Bespalov, D., Bai, B., Qi, Y., Shokoufandeh, A.: Sentiment classification based on supervised latent n-gram analysis. In: Proceedings of the 20th ACM International Conference on Information and Knowledge Management, pp. 375–382. ACM (2011)
12. Bespalov, D., Qi, Y., Bai, B., Shokoufandeh, A.: Sentiment lassification with supervised sequence embedding. In: Machine Learning and Knowledge Discovery in Databases, pp. 159–174. Springer (2012)
13. Hatzivassiloglou, V., Mckeown, K.R.: Predicting the semantic orientation of adjectives. In: Proceedings of ACL-97, 35th Annual Meeting of the Association for Computational Linguistics, pp. 174–181. Association for Computational Linguistics (1997)
14. Mao, Y., Lebanon, G.: Isotonic conditional random fields and local sentiment flow. Adv. Neural Inf. Process. Syst. **19**, 961 (2007)
15. Pang, B., Lee, L.: A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the 42nd annual meeting on Association for Computational Linguistics, p. 271. Association for Computational Linguistics (2004)
16. Wiebe, J.M.: Learning subjective adjectives from corpora. In: In AAAI, pp. 735–740 (2000)
17. Hatzivassiloglou, V., Wiebe, J.: Effects of adjective orientation and gradability on sentence subjectivity. In: Proceedings of the 18th Conference on Computational Linguistics, vol. 2, pp. 299–305. Universität des Saarlandes, Saarbrücken, Germany, July 31–Aug 4 (2000)
18. Wiebe, J., Mihalcea, R.: Word sense and subjectivity. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, pp. 1065–1072. Association for Computational Linguistics (2006)
19. Wiebe, J., Wilson, T., Bruce, R., Bell, M., Martin, M.: Learning subjective language. Comput. Linguist. **30**(3), 277–308 (2004)

20. Liu, B., Zhang, L.: A survey of opinion mining and sentiment analysis. In: Mining Text Data, pp. 415–463. Springer (2012)
21. Das, S.R., Chen, M.Y.: Yahoo! for amazon: sentiment extraction from small talk on the web. Manage. Sci. **53**(9), 1375–1388 (2007)
22. Turney, P.D., Littman, M.L.: Measuring praise and criticism: inference of semantic orientation from association. ACM Trans. Inf. Syst. (TOIS) **21**(4), 315–346 (2003)
23. Miller, G.A.: Wordnet: a lexical database for english. Commun. ACM **38**(11), 39–41 (1995)
24. Wilson, T., Wiebe, J., Hoffmann, P.: Recognizing contextual polarity: an exploration of features for phrase-level sentiment analysis. Comput. Linguist. **35**(3), 399–433 (2009)
25. Qiu, G., Liu, B., Bu, J., Chen, C.: Expanding domain sentiment lexicon through double propagation. In: Proceedings of the 21st international jont conference on Artifical intelligence, pp. 1199–1204 (2009)
26. Choi, Y., Cardie, C.: Adapting a polarity lexicon using integer linear programming for domain-specific sentiment classification. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pp. 590–598 (2009)
27. Dragut, E.C., Yu, C., Sistla, P., Meng, W.: Construction of a sentimental word dictionary. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, CIKM '10, pp. 1761–1764. ACM, New York, NY, USA (2010)
28. Lu, Y., Castellanos, M., Dayal, U., Zhai, C.: Automatic construction of a context-aware sentiment lexicon: an optimization approach. In: Proceedings of the 20th International Conference on World Wide Web, WWW '11, pp. 347–356. ACM, New York, NY, USA (2011)
29. Paltoglou, G., Gobron, S., Skowron, M., Thelwall, M., Thalmann, D.: Sentiment analysis of informal textual communication in cyberspace. Proc. Engage 13–25 (2010)
30. McDonald, R., Hannan, K., Neylon, T., Wells, M., Reynar, J.: Structured models for fine-to-coarse sentiment analysis. In: Annual Meeting-Association For Computational Linguistics, vol. 45, p. 432 (2007)
31. Kim, S.-M., Hovy, E.: Automatic detection of opinion bearing words and sentences. In: Proceedings of IJCNLP, vol. 5 (2005)
32. Wilson, T., Wiebe, J., Hoffmann, P.: Recognizing contextual polarity in phrase-level sentiment analysis. In: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, pp. 347–354. Association for Computational Linguistics (2005)
33. Meena, A., Prabhakar, T.: Sentence level sentiment analysis in the presence of conjuncts using linguistic analysis. In: Advances in Information Retrieval, pp. 573–580. Springer (2007)
34. Martineau, J., Finin, T.: Delta tfidf: an improved feature space for sentiment analysis. In: ICWSM (2009)
35. Salton, G., Wong, A., Yang, C.-S.: A vector space model for automatic indexing. Commun. ACM **18**(11), 613–620 (1975)
36. Denecke, K.: How to assess customer opinions beyond language barriers? In: ICDIM, IEEE, pp. 430–435 (2008)
37. Bifet, A., Frank, E.: Sentiment knowledge discovery in twitter streaming data. In: Discovery Science, pp. 1–15. Springer (2010)
38. Pak, A., Paroubek, P.: Twitter as a corpus for sentiment analysis and opinion mining. In: LREC (2010)
39. Zhang, E., Zhang, Y.: Ucsc on trec 2006 blog opinion mining. In: Text Retrieval Conference (2006)
40. Chang, C.-C., Lin, C.-J.: Libsvm: a library for support vector machines. ACM Trans. Intell. Syst. Technol. (TIST) **2**(3), 27 (2011)
41. Wang, H., Lu, Y., Zhai, C.: 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, pp. 783–792 (2010)
42. Bespalov, D., Qi, Y., Bai, B., Shokoufandeh, A.: Sentiment classification with supervised sequence embedding. In: Flach,P.A. Bie, T.D., Cristianini, N. (eds.) ECML/PKDD (1). Lecture Notes in Computer Science, vol. 7523, pp. 159–174. Springer (2012)

43. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. ACM SIGKDD Explor. Newsl. **11**(1), 10–18 (2009)
44. Esuli, A., Sebastiani, F.: Determining term subjectivity and term orientation for opinion mining. In: Proceedings of EACL, vol. 6, pp. 193–200 (2006)
45. Lau, R.Y.K., Lai, C.L., Bruza, P.B., Wong, K.F.: 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, CIKM '11, pp. 2457–2460. ACM, New York, NY, USA (2011)
46. Gindl, S., Weichselbraun, A., Scharl, A.: Cross-domain contextualisation of sentiment lexicons. In: Proceedings of the 19th European Conference on Artificial Intelligence (ECAI), 16 Aug 2010
47. Gezici, G., Yanikoglu, B., Tapucu, D., Saygın, Y.: New features for sentiment analysis: Do sentences matter?. In: SDAD 2012 The 1st International Workshop on Sentiment Discovery from Affective Data, p. 5 (2012)
48. Gräbner, D., Zanker, M., Fliedl, G., Fuchs, M.: Classification of customer reviews based on sentiment analysis. In: Information and Communication Technologies in Tourism 2012, pp. 460–470. Springer (2012)
49. Maas, A.L., Daly, R.E., Pham, P.T., Huang, D., Ng, A.Y., Potts, C.: Learning word vectors for sentiment analysis. In: Lin, D., Matsumoto, Y., Mihalcea, R. (eds.) ACL, pp. 142–150. The Association for Computer Linguistics (2011)