Sentiment Prediction Based on Dempster-Shafer Theory of Evidence

Article *in* Mathematical Problems in Engineering · April 2014

DOI: 10.1155/2014/361201

CITATIONS
15

READS
134

3 authors, including:

Ahmad Reza Naghsh-Nilchi
University of Isfahan
68 PUBLICATIONS **671** CITATIONS

SEE PROFILE

Nasser Ghasem-Aghaee
Sheikh Bahaei University
68 PUBLICATIONS **988** CITATIONS

SEE PROFILE

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 361201, 13 pages
http://dx.doi.org/10.1155/2014/361201

Research Article

Sentiment Prediction Based on Dempster-Shafer Theory of Evidence

Mohammad Ehsan Basiri, Ahmad Reza Naghsh-Nilchi, and Nasser Ghasem-Aghaee

Faculty of Computer Engineering & Information Technology, University of Isfahan, HezarJerib Avenue, Isfahan 81744, Iran

Correspondence should be addressed to Mohammad Ehsan Basiri; basiri@eng.ui.ac.ir

Received 1 January 2014; Accepted 5 March 2014; Published 27 April 2014

Academic Editor: Bo Shen

Sentiment prediction techniques are often used to assign numerical scores to free-text format reviews written by people in online review websites. In order to exploit the fine-grained structural information of textual content, a review may be considered as a collection of sentences, each with its own sentiment orientation and score. In this manner, a score aggregation method is needed to combine sentence-level scores into an overall review rating. While recent work has concentrated on designing effective sentence-level prediction methods, there remains the problem of finding efficient algorithms for score aggregation. In this study, we investigate different aggregation methods, as well as the cases in which they perform poorly. According to the analysis of existing methods, we propose a new score aggregation method based on the *Dempster-Shafer theory of evidence*. In the proposed method, we first detect the polarity of reviews using a machine learning approach and then, consider sentence scores as evidence for the overall review rating. The results from two public social web datasets show the higher performance of our method in comparison with existing score aggregation methods and state-of-the-art machine learning approaches.

1. Introduction

Online reviews are usually used by people who want to know other customers' opinion about products or services in which they are interested. These reviews are often provided in a free-text format by online review websites such as *Yelp* or *Amazon*. Although this rich source of information can help people decide better, they face the discouraging task of finding and reading numerous potential texts [1]. In order to make this task more convenient and time-efficient, some review websites provide average review ratings along with the text body of reviews.

The most common scheme of representing average review rating is five-star scores. Since these scores are only computable from numerical ratings, any review text should be first converted to a digit. There are two ways to do this: asking reviewers to suggest a numerical score for their review or using sentiment prediction techniques to calculate the overall rating of reviews from their text body. However, sentiment prediction methods are also used to extract other useful information from review texts. For example, the text body of reviews may be used to extract reviewers' opinion toward different aspects of a product [2], analyzing friendly exchanges in social network sites (SNSs) [3] or discussions of politics [4], and to find reviewers with suspicious behavior [5]. Therefore, it is important to design efficient sentiment prediction algorithms.

There are essentially two ways to use the text body of reviews for predicting their overall scores: considering any review as a unit of text or treating it as a collection of sentences, each with its own sentiment orientation and score. The main drawback of the first viewpoint is that it ignores the fine-grained structural information of textual content [2]. Therefore, the second point of view is preferable. In this approach, the sentiment score of each sentence within a review is first computed. Then, a score aggregation method is used to combine sentence-level scores into an overall review score. In fact, score aggregation can be seen as a data fusion step in which sentence scores are multiple sources of information that should be taken to generate a single review score.

Score aggregation is a widespread problem in sentiment analysis [1, 2, 6, 7]. Nevertheless, to our knowledge, no previous published research has investigated the role of aggregation method in sentiment prediction process. In fact, most of the existing approaches use simple rules of combination such as majority voting [6], aggregate-and-average [7], or scaling [1].

However, existing aggregation methods lack the accuracy required for sentiment prediction application. There are essentially two reasons for this. First, these methods usually do not consider all the important evidence that may be extracted from the text body of reviews. For example, most existing methods do not simultaneously consider the number of positive/negative words or sentences, the degree of positivity/negativity of words or sentences, and maximal agreements among these values. Second, existing methods do not use the polarity of reviews to decide about their rating. In other words, they consider rating prediction as a flat multiclass classification task.

In order to address these problems, we propose a hierarchical method based on the *Dempster-Shafer* (DS) theory of evidence [8]. In the proposed method, we first detect the polarity of reviews and then, using a DS-based aggregation method, predict review ratings. We choose DS theory since sentence scores can be considered as evidence for the overall review rating and DS is a traditional approach to evidential reasoning. Moreover, DS theory is a well-understood formal framework for combining multiple sources of evidence. Finally, by exploiting DS theory we can make use of all the available evidence to compute the overall review score.

We carried out our experiments on two large real-world social datasets with tens of thousands of reviews from popular review websites: CitySearch and TripAdvisor. These datasets contain reviews of various hotels and restaurants around the world.

The main contributions of this paper are as follows.

(i) We investigate the effect of applying different aggregation methods on sentence-level sentiment prediction and evaluate their performance on two large-scale and diverse social web datasets (Section 3.1).

(ii) We propose a hierarchical aggregation method based on the *Dempster-Shafer theory of evidence* and adapt it for sentiment prediction (Section 3.2).

(iii) We compare the accuracy of state-of-the-art machine learning techniques and the proposed system in review rating prediction (Section 4.3).

(iv) We adopt the proposed system for polarity detection and assess its performance on two datasets (Section 4.4).

The remainder of the paper is organized as follows. Section 2 reviews background and related work; Section 3 illustrates the materials and methods; Section 4 reports experimental results and presents a discussion of examined methods; finally Section 5 sets out conclusion and future work.

2. Literature Review

In this section, we first review related research on sentiment analysis. Then, we discuss sentiment strength detection and its applications.

2.1. Sentiment Analysis. Sentiment analysis is a subfield of data mining and natural language processing that deals with the extraction of people's opinions, evaluations, and emotions toward objects or events from their online reviews [9, 10]. Sentiment analysis and opinion mining refer to the same field of study [7]. Although the term opinion mining was first used by Dave et al. in [11] and the term sentiment analysis was first proposed by Nasukawa and Yi in [12], one can find their initial roots in the earlier works [13–15]. Since then, sentiment analysis has become a very active research area and, hence, there is a large body of research literature on this field. However, in this section we will not present a comprehensive review of the field, instead we just review some relevant researches. More comprehensive and detailed surveys can be found in [7, 9, 16].

Sentiment analysis has been receiving increasing attention recently from both academia and industry [16]. Some examples of sentiment analysis applications include predicting sales performance [17], ranking products and merchants [18], linking Twitter sentiment with public opinion polls [19], predicting election results [20], and identifying important product aspects [21]. In addition to these traditional applications, sentiment analysis has presented new research opportunities for social sciences in recent years [22–24]. For instance, finding how genders differed on emotional axes [25], predicting the stock market using Twitter moods [26], characterizing social relations [27], determining sentiment flow in social networks [28], and showing gender differences in giving and receiving positive sentiments in SNSs are some typical examples of social applications [29].

Sentiment analysis applications can be mainly investigated at three levels of granularity: document-level, sentence-level, and aspect-level [7]. The task at document level is to assign an overall sentiment orientation to the entire document. Sentence-level sentiment analysis, in turn, focuses on predicting the sentiment of sentences in isolation. Aspect-level techniques perform finer-grained analysis with the goal of discovering sentiments on entities and/or their aspects. The focus of this study is on the document-level sentiment prediction. In fact, our work may be considered as a sentiment strength detection task (see Section 2.2).

Sentiment analysis tasks can also be studied in terms of their sentiment prediction component. In this sense, existing approaches can be grouped into three main categories: machine learning approaches, linguistic-based strategies, and lexicon-based methods [6, 30]. In the first category, input texts are first converted to feature vectors and then, using a machine learning algorithm, a classifier is trained on a human-coded corpus. Finally, the trained classifier is used for sentiment prediction. Linguistic approaches, in turn, use simple rules based upon compositional semantics. In fact, they exploit the grammatical structure of text to predict its polarity. Lexicon-based approaches, however, work primarily

by identifying some predefined terms from a lexicon of known sentiment-bearing words. Then, an algorithm is used to predict the overall sentiment of a text based upon the occurrences of predefined words [6, 30, 31].

Most existing techniques for sentiment analysis use machine learning [7]. Machine learning approaches have some advantages such as the ability to identify the nonsentiment terms that carry implied sentiment by expressing a judgment (e.g., "cheap" in the phrase "this camera is cheap") [32]. Another advantage of these approaches is that any existing learning method such as support vector machines (SVM), Naïve Bayes, or artificial neural networks (ANNs) can be applied. However, there are some disadvantages with these methods, such as needing a corpus of human-coded texts for the training phase. In addition, although machine learning algorithms perform very well in the domain that they are trained on, their performance is suboptimal when applied to another domain [4, 30]. For example, in the domain of reviews about cell phones, the words "cheap" and "smart" are used to express positive opinion, while in the books domain the words "well researched" and "thriller" indicate positive sentiment. Therefore, an algorithm that is trained only on the cell phone domain may not correctly classify reviews from books domain. Moreover, as pointed out by Thelwall et al. in [32], algorithms of this category cannot "give a clear explanation as to why a sentence has been classified in a certain way, by reference to the predefined list of sentiment terms."

Linguistic-based methods are more suited when there are a lot of grammatically correct texts [33]. However, these approaches are of little use in cases with informal communication patterns. For example, ignoring the rules of grammar and spelling, by using emoticons (e.g., ":-D"), repeated letters or punctuation for emphasis (e.g., I miiiiiis you, or why?!!!), and texting-style abbreviations (e.g., "B4" instead of "before"), makes this approach less usable [6, 31].

Lexicon-based techniques use a list of sentiment-bearing words and phrases that is called opinion lexicon [7]. This lexicon is derived from the existing sources, such as the General Inquirer lexicon [34], the ANEW words [35], SentiWordNet [36], WordNet Affect [37], or the LIWC dictionary [38]. In addition to using these standard resources, some researchers have developed new methods to automatically generate and score lexicons [39, 40].

However, as pointed out by Liu in [7], opinion lexicon is necessary but not sufficient for sentiment analysis. Therefore, combined approaches are more useful. These approaches typically use supplementary information, such as semantic rules for dealing with negation [30], booster word list [6], emoticon list [31], and preexisting large collection of subjective common sense statement patterns [41]. According to Taboada et al., in [30], "lexicon-based methods for sentiment analysis are robust, result in good cross-domain performance, and can be easily enhanced with multiple sources of knowledge." Therefore, in the current study we propose a combined approach which works based on a lexicon-based method.

2.2. Sentiment Strength Detection. There are typically three common sentiment analysis tasks: subjectivity analysis with the aim of determining whether a given text is subjective or not [42, 43], polarity detection for assigning an overall positive or negative sentiment orientation to subjective texts [44], and sentiment strength detection which specifies the degree to which a text is positive or negative [6]. Subjectivity analysis and polarity detection have been investigated more than sentiment strength detection by sentiment analysis researchers. Therefore, there are a few algorithms for detecting sentiment strength in addition to sentiment polarity [31].

Sentiment strength detection can be formulated as a regression problem whereas the strength is expressed as ordinal rating scores [7]. For instance, Ganu et al. tried a linear regression model for review score prediction in [2] and later, in their recent work, they proposed a quadratic regression model that better fits their test dataset [1]. Nevertheless, some authors used other machine learning approaches to the problem. For example, Pang and Lee compared SVM regression with multiclass SVM and showed that the former is slightly better than the latter [45]. Afterwards, this approach was improved by Goldberg and Zhu by modeling the problem as a graph-based semisupervised learning problem [46].

Apart from machine learning approaches, some researchers have used lexicon-based or combined methods. As an early study, Neviarouskaya et al. proposed a combined method that uses a dictionary of terms and intensity ratings [47]. Recently, Thelwall et al. developed a lexicon-based approach called *SentiStrength* [48] and improved it later in [6]. The basic idea of these algorithms is that text often expresses the writer's emotional state and also we can differentiate between mild and strong emotions through our selected words [6, 31, 49, 50]. For example, "love" may be considered as a stronger positive word than "like" and, hence, it might be scored +4 by a sentiment strength detection algorithm, while "like" might be scored +2. These scores are then used by the algorithm to distinguish between weak and strong sentiment. Two recent and widely used lexicon-based tools for sentiment strength detection are SO-CAL [30] and SentiStrength [6].

SO-CAL (Semantic Orientation CALculator) uses lexicons of terms coded on a single negative to positive range of −5 to +5 [30]. Its opinion lexicon was built by human coders tagging lemmatized noun and verbs as well as adjectives and adverbs for strength and polarity in 500 documents from several corpora (e.g., rating "hate" as −4 and "hilariously" as +4). In SO-CAL intensifying words have a percentage associated with them. For example, "slightly" and "very" have −50% and +25% modification impacts, respectively. Therefore, if "good" has a sentiment value of 3, then "very good" would have a value of $3 \times (100\% + 25\%) = 3.75$. Moreover, it amplifies the strength of negative expressions in texts and decreases the strength of frequent terms. Empirical tests showed that SO-CAL has robust performance for sentiment detection across different types of web texts [30].

SentiStrength is a combined lexicon-based algorithm that uses additional linguistic information and rules for dealing with negation, misspelling, punctuations, and emoticons [48]. This classifier was built especially to cope with sentiment

detection in short informal English text [6]. The core of SentiStrength is a lexicon of 2,310 opinion words obtained from several sources [48]. The SentiStrength classifier's output is two values to each text: a measure of positive and a measure of negative sentiment, both on absolute integer scales ranging from one to five. Here, one signifies no sentiment and five denotes strong sentiment of each type. In this work we will use SentiStrength for sentence-level sentiment prediction for the following reasons. First, SentiStrength was designed to deal with social-media informal text, whereas SO-CAL was built for general text. Second, SentiStrength was designed to work at sentence-level, while SO-CAL was intended for determining document-level sentiment. Third, in addition to the lexicon and other word lists of SentiStrength (see Section 3.2), SentiStrength's Java version is also available.

3. Materials and Methods

In this section, we first compare the mechanism of different approaches to document-level sentiment prediction and then give an overview of our proposed system for predicting review scores. Finally, we present a detailed description of its important parts.

3.1. Document-Level Sentiment Prediction. As mentioned earlier, there are essentially two ways to use the text body of reviews for predicting their overall scores, considering any review as a unit of text or treating it as a collection of sentences, each with its own sentiment orientation and score. The main drawback of the first viewpoint is that it ignores the fine-grained structural information of textual content which may be extracted from sentences [2]. Therefore, the second point of view is preferable. An overview of these approaches is depicted in Figures 1 and 2, respectively.

In the current study, we follow the second approach. In this sense, our approach is similar to that of Ganu et al. [1], in which they modeled their problem as a multilabel text classification task for each sentence. However, there are two major differences between our approach and theirs. Firstly, our method is a document-level approach while their method works at aspect-level. Secondly, they modeled the prediction task as a regression problem and used a machine learning method while our approach is a lexicon-based method. An overview of our proposed system is depicted in Figure 3.

The input to our system in the training phase is a collection of reviews in the form of (*score*, *body*), where *score* is a reviewer-provided rating and *body* is the text content of the review. Note that these scores should not necessarily be provided by reviewers themselves. In such cases, a typical solution is to ask some human coders (annotators) to assign a score to each review based on its content [6, 31, 45]. However, in the testing phase, we only consider the text body of reviews and the output of the system is a five-star score for every test document.

The first module of the proposed system is *polarity detector* which classifies reviews into two classes: positive or negative. This module is used to provide additional information about each review to the aggregation mechanism which determines the overall review scores. We will discuss about this module in more detail in Section 4.5.

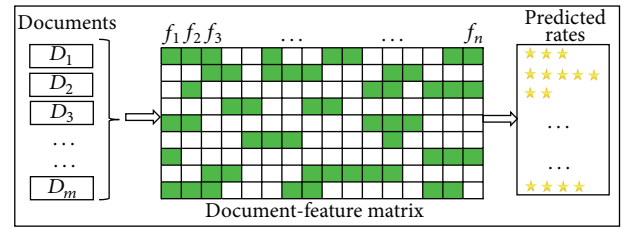

Figure 1: Considering any review as a unit of text for score prediction. In this view, the fine-grained structural information of textual content which may be extracted from sentences is not considered.

A sentence-level sentiment detection module is necessary in our system to detect the score of each sentence. We choose the *SentiStrength*, a free lexicon-based sentiment analysis software for classifying short informal web texts [31]. It seems to be suitable for our purpose, because it is designed for short informal text with abbreviations and slang that are common in online reviews. Furthermore, it has been tested in several domains and different online websites including MySpace, Twitter, BBC Forum, Digg.com, Runners World forum, and YouTube [6]. Reported results indicate that their performance is comparable with standard machine learning algorithms [6, 31, 48].

SentiStrength was used as a black box for sentence-level sentiment detection in our system. In fact, this package is used without any modifications on its scoring logic as the codes of the package are not publicly available. This module takes a sentence as input and returns its overall sentiment score as output. Therefore, before using this module each review should be decomposed into a set of sentences. Therefore, the second module of the flowchart in Figure 1 is the *sentence splitter*. It is a simple module for splitting a long review into independent sentences based on common sentence separators like colon and exclamation and question mark. The output of this module is a collection of sentences which are then passed to the next component. Because of the important role of the other modules in our system, we will describe them separately in the following subsections.

3.2. SentiStrength. SentiStrength is a lexicon-based sentiment strength detection software, developed by Thelwall et al. for social sentiment analysis of short informal texts [48]. To compute sentiment scores, SentiStrength uses the following lists:

(i) *sentiment strength word list* which is a collection of positive and negative terms, each with a value from 1 to 5;

(ii) *booster word list* that contains words for boosting or reducing the sentiment score of subsequent nonneutral words;

(iii) *idiom list* which is used to identify the sentiment of some common phrases;

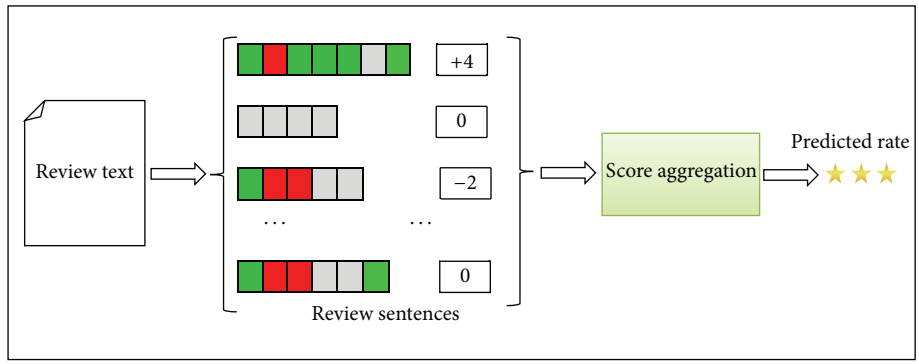

Figure 2: Considering any review as a collection of sentences. In this model, each sentence may have its own sentiment polarity and strength. The overall sentiment of document is computed using a score aggregation strategy that combines sentence-level scores.

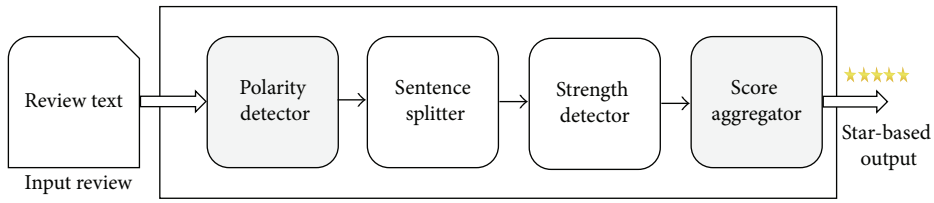

Figure 3: The overall view of modules used in the proposed system. First, the sentiment polarity of the input review is detected and then, using this information and sentence-level scores, an aggregation mechanism determines the overall review score.

(iv) *negation word list* that contains a few negation words for inverting subsequent sentiment words;

(v) *emoticon list* that supplements the first list. Some example words from these lists are shown in Table 1.

The output of SentiStrength may be shown in four ways: binary, trinary, scale, and default mode. In the binary mode, the polarity of input text is specified with +1 or −1 scores for positive and negative texts, respectively. In the trinary mode, the software reports positive/negative/neutral classification instead (with +1, −1, and 0, resp.). In the scale mode, the output of the software ranges from −4 to +4. The default mode is used unless binary, trinary, or scale mode is selected and it ranges from +1 (neutral) to +5 (extremely positive) for positive text and from −1 (neutral) to −5 (extremely negative) for negative texts. However, in the scale or default mode, the output is in the form of $\langle P, N\rangle$, where P and N are the maximum positive and the minimum negative scores of constituent words, respectively [48]. Table 2 shows some examples of how SentiStrength generates output.

3.3. Score Aggregation Strategies. In order to calculate the overall sentiment score of a long review based on its sentence ratings, an aggregation mechanism is needed. Score aggregation can be seen as a data fusion step in which sentence scores are multiple sources of information that should be taken to generate a single review score. Although score aggregation plays an important role in sentiment prediction applications, there are few works which directly address the problem. Some of the most important existing methods are as follows.

Table 1: Sample words from the lexicon of SentiStrength.

List name	Sample word	Score
Sentiment strength word list	Awful	−4
	Blissful	+5
Booster word list	Slightly	−1
	Extremely	+2
Idiom list	Shock horror	−2
	What's good	+2
Negation word list	Can't	—
	Never	—
Emoticon list	:'(	−1
	:-D	+1

SentiStrength. To calculate the overall sentiment of a long text, *SentiStrength* first computes the overall positive and negative sentiment of each sentence. Then, the maximum of positive and the minimum of negative sentence scores are taken. Consider the following example.

Example 1. The legacy suite was beautiful. The room provided an excellent view and had a great location. Room service was bad but the food was nice.

This text would be classified as follows: "The legacy suite was beautiful [3] ⟨sentence: 3, −1⟩. The room provided an excellent [4] view and had a great [3] location ⟨sentence: 4, −1⟩. Room service was bad [−2] but the food was nice [2] ⟨sentence: 2, −2⟩." While numbers in square brackets

Table 2: Some typical examples to show how SentiStrength works.

Input sentence	Sentiment words	Positive score	Negative score	Binary score	Trinary output	Scale output
In their book, the authors spell out the program, week-by-week.	—	+1	−1	+1	0	0
This book is beautifully written.	Beautifully [3]	+3	−1	+1	+1	+2
I like this book.	Like [2]	+2	−1	+1	+1	+1
I was watching a poorly made movie.	Poorly [−2]	+1	−2	−1	−1	−1
Why would I want to waste my time on what is unfortunately complete trash?	Waste [−2] Unfortunately [−2] Complete trash [−2][−1 booster word]	+1	−3	−1	−1	−2
I really love the book but hate the author.	Love [3] [+1 booster word] Hate [−4]	+4	−4	+1	−1	0

indicate the score of the preceding word, angle brackets show sentence scores, and the overall scores are +4 and −2.

Maximum of Scores. In this method, the maximum of the absolute values of the positive/negative scores is taken as the overall score of a review. However, this is not an appropriate strategy when, for example, a review contains several weak positive sentences and a strong negative sentence. For instance, consider the following review about a restaurant.

Example 2. It was the cheapest I found in the area and close to my apartment. The location is easy to find. The customer service was quick and friendly. The food was fresh and quite good. The decor is cute and intimate inside. The only drawback was that they only take cash or check and it was really awful.

Although it should be considered as a positive review, using *maximum of scores* method this review is assigned a −5 score. The reason is that each of its first five sentences has, at most, a score of +2 and the last sentence has a score of −5.

Scaled Rate. This strategy uses the number of positive and negative sentences to compute the final review score, ignoring the strength of sentences, as the following equation depicts [1, 2]:

$$\text{Aggregated Score} = \left(\frac{P}{P+N} * 4\right) + 1, \tag{1}$$

where P and N are the number of positive and negative sentences in the review, respectively. The major drawback of this method is that it does not consider the strength of sentiment. For example, consider the following two short reviews.

Example 3. Great tasting pizza, excellent music and environment.

Example 4. The food was good, music and environment was not so bad.

According to (1), both reviews will be considered as five-star reviews, because in both cases $P = 1$, and $N = 0$. However, it is clear that the former is more positive than the latter.

Sum of Predictions. This strategy is proposed to increase the accuracy of document-level sentiment classification, using additional information from paragraph-level annotations [51]. In this method, the probabilities of each paragraph belonging to positive, negative, and neutral classes are computed. Then, the probability scores for each class are summed across all paragraphs of a document. Finally, the class with the highest score is selected.

Sum of Maximums. This method is similar to the *maximum of scores* strategy in that it uses maximum of positive and negative scores. However, instead of selecting one with greater absolute value, it simply adds maximum of positive and negative scores. Therefore, the aggregated score for Example 1 using this strategy would be −3, while using *maximum of scores* results in a score of −5.

Majority Voting. This method is also used in [51] for paragraph aggregation and simply takes a majority vote of the probabilities generated for the paragraphs of a document. As pointed out by Ferguson et al., in [51], *sum of predictions* method outperforms majority voting in 3-point document classification (positive, negative, and neutral classes). However, this strategy has the same drawback of *maximum of score* strategy.

SimAvg. This strategy along with the next two algorithms is proposed in [49] for product review aggregation. In these three strategies, the overall score of a product is computed in an aggregation mechanism, such as the following equation:

$$V(P) = \frac{\sum_{i=1}^{n} \left(u\left(T_i(P)\right) * \text{Polarity}\left(T_i(P)\right)\right)}{\sum_{i=1}^{n} u\left(T_i(P)\right)}, \tag{2}$$

where V is the overall valuation (score) of a product P. It can be computed as a weighted average of the polarity

of each individual review $T_i(P)$ where the weights indicate their relative usefulness [49]. *SimAvg* method computes the aggregated polarity of a product by replacing $u(T_i(P))$ with 1 in (2). In other words, it computes the overall polarity as a simple average of individual review polarities. Although it seems a fair mechanism, we will show that using this strategy does not result in high accuracy in our experiments.

PredAvg. This method computes the overall polarity based on the weighted averaging mechanism of (2), where the weight assigned to each review T_i is its calculated usefulness score, $\widehat{u}(T_i(P))$. As pointed out in [49], this score can be computed with a learning algorithm such as support vector regression (SVR) using features that can distinguish useful and useless reviews.

GsAvg. This strategy computes the aggregated polarity based on (2) by replacing $u(T_i(P))$ with gold-standard usefulness score of review. Zhang et al. in [52] proposed to compute this score by dividing number of people who found this review useful by total number of reviewers. The results showed that *GsAvg* always outperforms *SimAvg,* while *PredAvg* in most cases falls between the two.

However, most of the described aggregation strategies neither were designed for score aggregation, nor have any theoretical basis. To fill this gap, we propose a new aggregation mechanism based on the *Dempster-Shafer theory of evidence* [8] and apply it to the results obtained from the SentiStrength module (see Figure 1).

3.4. Dempster-Shafer Theory of Evidence. In this section, we first present a brief description of Dempster-Shafer (DS) theory and then describe the way in which we apply it to the score aggregation problem. Dempster-Shafer is a theory of uncertainty that helps to quantify the degree to which some source of evidence supports a particular proposition. In fact, it is an alternative to traditional probability theory, allowing the explicit representation of ignorance and combination of evidence. This theory was originally developed by Dempster [50] and then extended by Shafer in his 1976 book, *A Mathematical Theory of Evidence* [8].

In DS theory, a problem domain can be specified by a finite set θ of mutually exclusive hypotheses, called *frame of discernment*. Moreover, a *mass function* ($m(A)$) is used to represent the strength of evidence supporting a subset $A \subseteq \theta$ based on a given piece of evidence [53]. In other words, a mass function is a *basic probability assignment* (BPA) to all subsets X of θ. This function returns a real number in the range [0, 1] and has the following properties:

$$m(\emptyset) = 0, \qquad \sum_{A \in 2^{\theta}} m(A) = 1, \tag{3}$$

where $m(A)$ can be interpreted as the belief exactly committed to A, based on the available evidence. A subset A of θ is called a *focal element* of a mass function m over θ, if $m(A) > 0$.

The fundamental operation of DS theory of evidence is a rule for the pooling of evidence from a variety of sources, known as *Dempster's rule of combination* [8]. Specifically, this rule has been proposed for aggregating two independent bodies of evidence over a common frame of discernment into one body of evidence. Moreover, this rule can be used to aggregate new evidence with old evidence based on data from both new and old evidence. Given two pieces of evidence on the same frame θ represented by two BPAs, m_1 and m_2, Dempster's rule of combination (also called the orthogonal sum of m_1 and m_2 and denoted by $m_1 \oplus m_2$) is defined as follows:

$$m_{1,2}(A) = \frac{\sum_{X \cap Y = A} m_1(X)\, m_2(Y)}{1 - \sum_{X \cap Y = \emptyset} m_1(X)\, m_2(Y)}, \tag{4}$$

where the denominator is a normalization factor to ensure that $m_{1,2}$ remains a BPA. Moreover, since

$$\sum_{X \cap Y = A} m_1(X)\, m_2(Y) = \sum_{X \cap Y = A} m_1(Y)\, m_2(X), \tag{5}$$

we can write

$$\begin{aligned} m_{1,2}(A) &= \frac{\sum_{X \cap Y = A} m_1(X)\, m_2(Y)}{1 - \sum_{X \cap Y = \emptyset} m_1(X)\, m_2(Y)} \\ &= \frac{\sum_{X \cap Y = A} m_2(X)\, m_1(Y)}{1 - \sum_{X \cap Y = \emptyset} m_2(X)\, m_1(Y)} \\ &= m_{2,1}(A). \end{aligned} \tag{6}$$

Therefore, Dempster's rule of combination is commutative. Furthermore, if we define $d_{1,2} = \sum_{X \cap Y = \emptyset} m_2(X) m_1(Y)$, then

$$m_{(1,2),3}(A) = \frac{\sum_{X \cap Y = A} \left(\sum_{U \cap V = X} m_1(U)\, m_2(V) / (1 - d_{1,2}) \right) m_3(Y)}{1 - \sum_{X \cap Y = \emptyset} \left(\sum_{U \cap V = X} m_1(U)\, m_2(V) / (1 - d_{1,2}) \right) m_3(Y)}, \tag{7}$$

$$m_{(1,2),3}(A) = \frac{\sum_{U \cap V \cap Y = A} m_1(U)\, m_2(V)\, m_3(Y)}{1 - \sum_{U \cap V \cap Y = \emptyset} m_1(U)\, m_2(V)\, m_3(Y)}, \tag{8}$$

$$\begin{aligned} m_{(1,2),3}(A) &= \frac{\sum_{X \cap Y = A} \left(\sum_{U \cap V = Y} m_1(U)\, m_2(V) / (1 - d_{2,3}) \right) m_1(X)}{1 - \sum_{X \cap Y = \emptyset} \left(\sum_{U \cap V = Y} m_1(U)\, m_2(V) / (1 - d_{2,3}) \right) m_1(X)} \\ &= m_{1,(2,3)}(A). \end{aligned} \tag{9}$$

Thus, from (6) and (9) we can conclude that the result of applying Dempster's rule of combination is independent of the order in which the evidence is combined. For example, in the case of three pieces of evidence, we have the following rule of combination:

$$m_{1,2,3}(A) = \frac{\sum_{X \cap Y \cap Z = A} m_1(X)\, m_2(Y)\, m_3(Z)}{1 - \sum_{X \cap Y \cap Z = \emptyset} m_1(X)\, m_2(Y)\, m_3(Z)}. \tag{10}$$

In general, higher number of evidence can be dealt with similarly. Specifically, given n distinct BPAs, m_i for $i \in \{1, 2, \ldots, n\}$, Dempster's rule of combination is [51]

$$m_{1,\ldots,n}(A) = \frac{\sum_{\bigcap_{i=1}^{n} X_i = A} \left(\prod_{j=1}^{n} m_j(X_i) \right)}{1 - \sum_{\bigcap_{i=1}^{n} X_i = \emptyset} \left(\prod_{j=1}^{n} m_j(X_i) \right)}. \tag{11}$$

This can be proved by induction on n. Clearly, for $n = 2$, as shown in (4), it is true. Assume that this is true for $n = t-1$. Then, we have

$$m_{1,\ldots,t-1}(A) = \frac{\sum_{\bigcap_{i=1}^{t-1} X_i = A} \left(\prod_{j=1}^{t-1} m_j(X_i)\right)}{1 - \sum_{\bigcap_{i=1}^{t-1} X_i = \emptyset} \left(\prod_{j=1}^{t-1} m_j(X_i)\right)}. \quad (12)$$

Now to show that it is true for $n = t$, we can replace $m_{(1,2)}(A)$ by $m_{1,\ldots,t-1}(A)$ and $m_3(A)$ by $m_t(A)$ in (7) and follow the same steps we used to prove (9).

The first step to apply DS theory to our score aggregation problem is to define evidence based on the output of the SentiStrength module of proposed system (see Figure 1). As mentioned earlier, we use SentiStrength for sentence-level sentiment detection. In this work, we consider each sentence score as evidence for the overall score of a review. This seems reasonable, since it has been shown that not only sentence-level evidence can be aggregated for document-level polarity prediction, but also more effective sentiment prediction methods can be designed using a sentence-level approach [52].

Having defined evidence in our context, the next step is to specify the mass function. To this aim, we suggest to use normalized sentence score according to the following equation:

$$m(S) = \frac{sentenceScore - minScore}{maxScore - minScore}, \quad (13)$$

where S is a sentence, *sentenceScore* is the output of SentiStrength for this sentence, and *maxScore* and *minScore* are the maximum and minimum scores of positive and negative sentences, respectively. This mass function expresses the degree of positivity of S, and it is straightforward to show that it satisfies the conditions given in (3). Note that the output of this mass function may also be interpreted as the probability of being a five-star review.

There are two reasons behind the choice of this mass function. First, it is natural to map sentiment strength prediction task to a binary classification problem and, hence, to measure the degree of positivity of reviews [7]. Second, a similar equation has been proposed by Wang et al. in [54] and successfully applied to the problem of text categorization. In fact, their mass function is a *categorization-specific* function and as they stated, "[it] expresses the degrees of beliefs in respective propositions corresponding to each category to which a given document could belong" [54]. Specifically, this mass function is

$$1 \le i \le |C|: \; m(c_i) = \frac{w(c_i)}{\sum_{k=1}^{|C|} w(c_k)}, \quad (14)$$

where each $c_i \in C$ is a proposition of the form "*document d belongs to category c_i*" and C is the frame of discernment. Moreover, there is a function $\varphi(d)$ for assignment of categories to d in the form of

$$\varphi(d) = \left(m(c_1), m(c_2), \ldots, m(c_{|C|})\right). \quad (15)$$

This function can be seen as "a piece of evidence that indicates the strength of our confidence that the document comes from each category" [54].

However, our problem is different from that of Wang et al. in several aspects. Firstly, we do not consider sentiment strength detection as a multiclass classification task. Secondly, our proposed method, unlike that of [54], is a sentence-level procedure. Thirdly, we do not use several classifiers for sentiment detection of individual sentences within reviews. Fourthly, we consider the polarity of each sentence of a review as evidence for the overall score of it, while in [54] each classifier output is considered as evidence for the final decision about the category to which the document belongs.

The final step in applying DS theory in our system is to use Dempster's rule of combination for aggregating sentence scores into the overall review score. This could be done by using (11), where n is the number of nonneutral sentences of a review. To avoid the computational complexity of (11), we can iteratively apply (4). In this way, (4) can be rewritten as

$$m(A) = \frac{\sum_{X \cap Y = A} m_n(X)\, m_o(Y)}{1 - \sum_{X \cap Y = \emptyset} m_n(X)\, m_o(Y)}, \quad (16)$$

where m_n and m_o are measures of confidence from new and old existing evidence, respectively. The old evidence is the mass m from the previous iteration of Dempster's rule of combination. Consider the following example.

Example 5. The food was eclectic and delicious. The service is possibly the worst ever, and the music is on so loud that you are unable to maintain a conversation. As soon as I paid my bill I was asked to leave. That was rude and left an unpleasant impression on me. The restaurant was very bad in terms of decor.

The iterative process of score aggregation according to (16) has been shown in Table 3.

The overall score of a review can be computed by converting the aggregated mass m (e.g., 0.1 in the above example) to a five-star score. This can be simply done by using the following equation:

$$\text{Score} = \text{round}\left((m \times 4) + 1\right). \quad (17)$$

4. Results and Discussion

In the previous section, we have described the proposed sentiment prediction system. In this section, we first describe briefly the datasets used in this work and then present empirical evaluation results. Moreover, we aim at addressing the following research questions.

(1) Are complex methods necessary for aggregating sentence scores into document ratings?

(2) Can applying Dempster-Shafer theory of evidence to score aggregation improve review rating prediction?

(3) Can the use of document polarities in a hierarchical manner improve the performance of review rating prediction?

Table 3: The process of applying Dempster's rule of combination to Example 5.

Iteration number	Sentence score	$m(A)$	Aggregated m
1	+2	0.75	0.75
2	−2	0.25	0.5
3	0	0.5	0.5
4	−2	0.25	0.25
5	−2	0.25	0.1

(4) Can the proposed aggregation method be applied to polarity detection?

4.1. Datasets and Evaluation Metrics. We evaluate the performance of our proposed method on the following reviewer-coded social datasets (i.e., each review has been quantified with a five-star score by its writer).

(i) *TripAdvisor dataset*: this dataset contains customer reviews for various hotels across the world [54]. The *TripAdvisor* dataset is one of the largest sentiment classification datasets currently available [55].

(ii) *CitySearch restaurant review dataset*: this dataset, introduced in [2], was extracted from the NY CitySearch web site. As pointed out in [1], this dataset is sparse and over 25,000 reviewers have written only one review [1].

For evaluation, we used two measures: mean absolute error (MAE) and accuracy. MAE is defined as the mean of the absolute differences between the predicted and the reviewer-provided scores. The accuracy of a prediction method may be calculated according to the following equation:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}, \tag{18}$$

where TP and TN are true positive and true negative, while FP and FN are false positive and false negative, respectively.

4.2. The Comparison of Aggregation Methods. In this section, we describe the experiment which has been performed to evaluate different score aggregation methods given in Section 3.3. In order to show the effectiveness of aggregation methods in this and the following sections, the results for DS algorithm are obtained independently of the *polarity detector* module. Tables 4 and 5 summarize the performance of aggregation methods on two review collections in terms of the MAE and accuracy. As pointed out in Section 3, some of the described algorithms were not designed for score aggregation (e.g., *PredAvg* and *GsAvg*) and, hence, we could not compare them with the proposed method.

The first row of both tables shows the results of using a simple algorithm that assigns a random integer in the range $[1, 5]$ to each review. Other algorithms have been described earlier in Section 3.3. *SimAvg* is the simplest aggregation method that is used in the absence of other mechanisms. An interesting result in Tables 4 and 5 is that the performance

Table 4: The comparison of the MAE of different aggregation methods.

Method/dataset	CitySearch	TripAdvisor
Random	1.49	1.48
SimAvg	1.89	2.01
Sum of maximums	0.97	0.95
Maximum of scores	1.07	1.10
Dempster-Shafer	**0.89**	**0.90**
Scaled rate	1.46	1.39

Table 5: The comparison of the accuracy of different aggregation method.

Method/dataset	CitySearch	TripAdvisor
Random	0.50	0.49
SimAvg	0.48	0.46
Sum of maximums	0.63	0.51
Maximum of scores	0.62	0.63
Dempster-Shafer	**0.72**	**0.71**
Scaled rate	0.55	0.56

of the *SimAvg* method is lower than the random approach in terms of both MAE and accuracy. This shows that simple averaging is not an appropriate strategy for review score aggregation. Therefore, we have successfully addressed our first research question: using more complex aggregation methods is necessary in computing final review scores.

It can be seen that in all cases, *Dempster-Shafer* method outperforms other aggregation algorithms. It seems that there are at least two reasons for *Dempster's rule of combination* to outperform the other aggregation methods. First, it considers all sentences' scores (as pieces of evidence) in computing the overall score of a document. Second, it takes account of maximal agreements among these pieces of evidence. However, none of the above-discussed aggregation methods consider these two factors simultaneously. Going back to our second research question, these results show that applying Dempster-Shafer aggregation method to sentence-level sentiment score combination improves review rating prediction.

4.3. Comparing the Proposed System and Machine Learning Approaches. To assess the effectiveness of the proposed system, we compare it with frequently used machine learning algorithms: Ada-Boost, Bayesian-Net, decision tree (J48 classification), K-Star, Naïve Bayes, and Support Vector Machines (Sequential Minimal Optimization variant, SMO). These algorithms were implemented using *TagHelper*, a text analysis program written on the top of the *Weka* machine learning software [56]. A training model was built using common features including punctuation, unigrams, bigrams, part-of-speech, and line length. Moreover, all features occurring less than five times were removed and stemming was used to allow some forms of generalization across lexical items. The MAE and accuracy values of the mentioned approaches are compared in Figures 4 and 5, respectively.

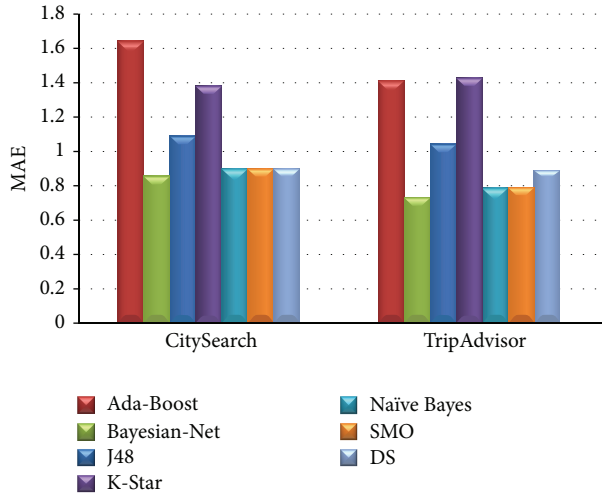

Figure 4: The comparison of the MAE of machine learning algorithms and the proposed DS-based approach on CitySearch and TripAdvisor datasets.

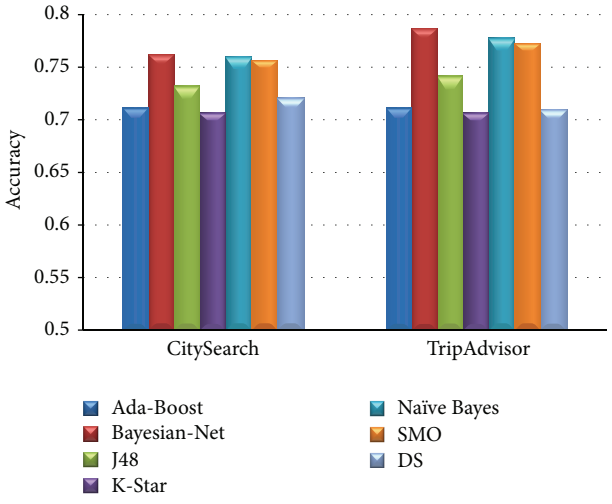

Figure 5: The comparison of the accuracy of machine learning algorithms and the proposed DS-based approach on CitySearch and TripAdvisor datasets.

Although the proposed method does not perform as well as some of machine learning algorithms, the difference is negligible. As pointed out in [6], the higher performance of machine learning algorithms can be the result of selecting indirect features like topic or discourse features rather than features that directly identify the sentiment. However, the main advantage of the proposed system is that it outperforms machine learning approaches in terms of speed and memory requirements. For example, on CitySearch dataset, machine learning algorithms need several minutes, on average, for training while the proposed method detects the scores in a few seconds on a standard PC with 6 GB RAM and 2.5 GHz Intel Core i5 CPU. Moreover, we could not use many machine learning algorithms on these datasets, possibly due to the limitations of Weka in terms of processing resources. It should be noted that even with a larger amount of RAM, some machine learning algorithm could not be used on large datasets. For example, Thelwall et al. report that "Logistic Regression did not complete a single evaluation (out of 30) on 1000 features within two weeks" on a 96 GB machine [6]. In summary, the proposed lexicon-based approach can achieve comparable accuracy with lower time and space complexity.

4.4. Applying the Proposed System to Polarity Detection. To answer the third research question, we apply the proposed Dempster-Shafer aggregation method to polarity detection problem. In this case, each review should be classified as positive or negative. Therefore, we considered each four- or five-star review as positive and each one- or two-star review as negative. In this study, we did not consider neutral class and, hence, three-star reviews were ignored. Experiments were carried out on two previously described review collections and the obtained results are shown in Tables 6 and 7.

There are two notable results in Tables 6 and 7. First, the overall performance of all aggregation methods in these tables is higher than their performance in rating prediction case (i.e., Tables 4 and 5). This is expected since our approach is lexicon-based and in positive/negative case, sentiment words can be more accurately specified than in five-star case. For example, consider the following review.

Example 6. Very nice hotel, I stay here for work on occasions. The rooms are lovely and the service is always friendly. The beds are extremely comfy.

This is, of course, a positive review and is classified correctly by all described methods. However, most algorithms classify it as five-star, while it is actually a four-star review. A similar problem exists for negative reviews.

The second observation is that in all cases the proposed approach outperforms other aggregation mechanisms in terms of MAE and accuracy. Therefore, the third question was also answered: the proposed Dempster-Shafer aggregation method may be successfully applied to polarity detection problem.

4.5. The Effect of Using Polarity Detector Module. As mentioned earlier, in order to further improve the performance of the proposed system, we combine our lexicon-based approach with machine learning methods in a hierarchical manner. Specifically, we first use the *SMO* algorithm to detect the polarity of each review. Then, the rating of each review is predicted by our proposed *Dempster-Shafer* based method, considering the results of the previous step. In other words, the first step classifies reviews as positive or negative, and the second step further classifies positive (negative) reviews as either four- or five-star (one- or two-star). The comparison of this hierarchical approach (SMO-DS) with DS and SMO methods on CitySearch and TripAdvisor datasets is shown in Figures 6 and 7, respectively.

As can be seen in Figures 6 and 7, the hierarchical approach outperforms both lexicon-based and machine learning-based methods in terms of MAE. Moreover, the

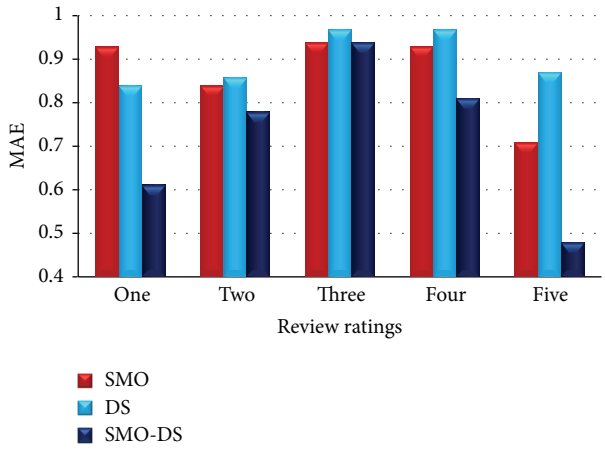

Figure 6: The comparison of the MAE of SMO, DS, and SMO-DS algorithms on CitySearch dataset.

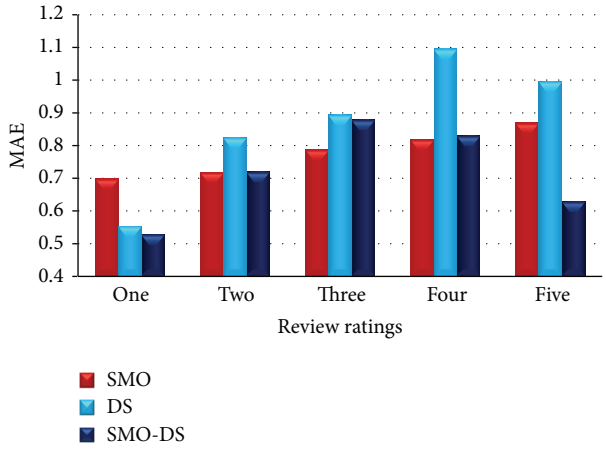

Figure 7: The comparison of the MAE of SMO, DS, and SMO-DS algorithms on TripAdvisor dataset.

improvement is greater for one- and five-star reviews which are very negative and very positive reviews, respectively. Another notable result is that for three-star reviews the hierarchical approach makes no improvement. This is expected because three-star reviews are neutral and the polarity detection step adds no significant information to the rating prediction process.

5. Conclusions and Future Work

In this paper, we studied the impact of score aggregation methods on the accuracy of sentiment prediction systems. Specifically, we studied different methods to aggregate sentence-level sentiment scores into a document rating and discussed their advantages and disadvantages. Then, we proposed an approach based on the *Dempster-Shafer theory of evidence*. In the proposed system, we first detect the polarity of each review and then predict the overall review score based on the results of previous step. In order to predict review scores, we first detect the individual sentences' scores

Table 6: The comparison of the MAE of different aggregation methods for polarity detection.

Method/dataset	CitySearch	TripAdvisor
Random	0.55	0.44
SimAvg	0.68	0.55
Sum of maximums	0.41	0.30
Maximum of scores	0.33	0.26
Dempster-Shafer	**0.22**	**0.19**
Scaled rate	0.44	0.44

Table 7: The comparison of the accuracy of different aggregation methods for polarity detection.

Method/dataset	CitySearch	TripAdvisor
Random	0.44	0.55
SimAvg	0.32	0.45
Sum of maximums	0.58	0.69
Maximum of scores	0.67	0.74
Dempster-Shafer	**0.77**	**0.81**
Scaled rate	0.56	0.65

within a review and then aggregate them into a five-star review score. For sentence-level sentiment detection, we used *SentiStrength*, an available lexicon-based sentiment strength detection library.

Moreover, we compared our lexicon-based approach with state-of-the-art machine learning algorithms, namely, AdaBoost, Bayesian networks, decision tree (J48 classification), K-Star, Naïve Bayes, and Support Vector Machines (sequential minimal optimization variant (SMO)). Experiments were carried out on two available real-world social datasets with tens of thousands of reviews from popular online review sites: CitySearch and TripAdvisor. We found that the proposed Dempster-Shafer based approach outperforms existing aggregation methods in terms of accuracy and MAE. Also, we found that our proposed method is not only applicable for review polarity detection, but also it outperforms other aggregation methods for polarity detection.

The main novel contributions of current study are as follows: showing the necessity of using aggregation mechanism for review rating prediction; discussing the existing aggregation methods and evaluating their performance on two large-scale and diverse social web datasets; comparing the accuracy of state-of-the-art machine learning techniques and the proposed lexicon-based approach for sentiment prediction; and designing an aggregation method based on *Dempster-Shafer theory of evidence* and adapting it for sentiment prediction.

A promising direction for future work would be investigating other mathematical theories to aggregate sentence-level evidence into review-level polarity measures (e.g., Bayesian data fusion and order weighted averaging operators). Our approach would be appropriate for other tasks in sentiment analysis such as emotion detection. As pointed out by Luneski et al., in [57], there is a significant motivation for investigating emotions and their impact on human health.

Moreover, fine-grained emotion detection can increase the effectiveness of sentiment analysis applications. Therefore, adapting the proposed system to detect emotion from social text could also be investigated in future research.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This research was partially supported by Iran Telecommunication Research Center (contract T5008801).

References

[1] G. Ganu, Y. Kakodkar, and A. Marian, "Improving the quality of predictions using textual information in online user reviews," *Information Systems*, vol. 38, pp. 1–15, 2013.

[2] G. Ganu, N. Elhadad, and A. Marian, "Beyond the stars: improving rating predictions using review text content," *WebDB*, 2009.

[3] G. Paltoglou and M. Thelwall, "Twitter, MySpace, Digg: unsupervised sentiment analysis in social media," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 3, article 66, 2012.

[4] H. Stuckenschmidt and C. Zirn, "Multi-dimensional analysis of political documents," in *Natural Language Processing and Information Systems*, pp. 11–22, Springer, 2012.

[5] T. Lappas, "Fake reviews: the malicious perspective," in *Natural Language Processing and Information Systems*, pp. 23–34, Springer, 2012.

[6] M. Thelwall, K. Buckley, and G. Paltoglou, "Sentiment strength detection for the social web," *Journal of the American Society for Information Science and Technology*, vol. 63, no. 1, pp. 163–173, 2012.

[7] B. Liu, *Sentiment Analysis and Opinion Mining*, vol. 5 of *Synthesis Lectures on Human Language Technologies*, 2012.

[8] G. Shafer, *A Mathematical Theory of Evidence*, vol. 1, Princeton University Press, Princeton, NJ, USA, 1976.

[9] B. Liu, "Sentiment analysis: a multifaceted problem," *IEEE Intelligent Systems*, vol. 25, no. 3, pp. 76–80, 2010.

[10] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Foundations and Trends in Information Retrieval*, vol. 2, no. 1-2, pp. 1–135, 2008.

[11] K. Dave, S. Lawrence, and D. 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*, pp. 519–528, 2003.

[12] T. Nasukawa and J. Yi, "Sentiment analysis: capturing favorability using natural language processing," in *Proceedings of the 2nd international conference on Knowledge capture*, pp. 70–77, 2003.

[13] J. Wiebe, "Learning subjective adjectives from corpora," in *Proceedings of the 7th National Conference on Artificial Intelligence and 12th Conference on Innovative Applications of Artificial Intelligence (AAAI/IAAI '00)*, pp. 735–740, 2000.

[14] R. M. Tong, "An operational system for detecting and tracking opinions in on-line discussions," in *Proceedings of Workshop on Operational Text Classification*, The Working Notes of the ACM SIGIR, pp. 1–6, 2001.

[15] S. Das and M. Chen, "Yahoo! for Amazon: extracting market sentiment from stock message boards," in *Proceedings of the Asia Pacific Finance Association Annual Conference (APFA '01)*, 2001.

[16] H. Chen and D. Zimbra, "AI and opinion mining," *IEEE Intelligent Systems*, vol. 25, no. 3, pp. 74–76, 2010.

[17] Y. Liu, X. Huang, A. An, and X. Yu, "ARSA: a sentiment-aware model for predicting sales performance using blogs," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*, pp. 607–614, July 2007.

[18] M. McGlohon, N. S. Glance, and Z. Reiter, "Star quality: aggregating reviews to rank products and merchants," in *Proceedings of the International Conference on Weblogs and Social Media (ICWSM '10)*, 2010.

[19] B. O'Connor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith, "From tweets to polls: linking text sentiment to public opinion time series," in *Proceedings of the International Conference on Weblogs and Social Media (ICWSM '10)*, vol. 11, pp. 122–129, 2010.

[20] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe, "Predicting elections with twitter: what 140 characters reveal about political sentiment," in *Proceedings of the International Conference on Weblogs and Social Media (ICWSM '10)*, vol. 10, pp. 178–185, 2010.

[21] J. Yu, Z.-J. Zha, M. Wang, and T.-S. Chua, "Aspect ranking: Identifying important product aspects from online consumer reviews," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT '11)*, pp. 1496–1505, June 2011.

[22] A. Minamikawa and H. Yokoyama, "Personality estimation based on weblog text classification," in *Modern Approaches in Applied Intelligence*, pp. 89–97, Springer, 2011.

[23] J. Bollen, B. Gonçalves, G. Ruan, and H. Mao, "Happiness is assortative in online social networks," *Artificial Life*, vol. 17, no. 3, pp. 237–251, 2011.

[24] G. Miller, "Social scientists wade into the tweet stream," *Science*, vol. 333, no. 6051, pp. 1814–1815, 2011.

[25] S. M. Mohammad and T. W. Yang, "Tracking sentiment in mail: how genders differ on emotional axes," in *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (ACL-HLT '11)*, pp. 70–79, 2011.

[26] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of Computational Science*, vol. 2, no. 1, pp. 1–8, 2011.

[27] G. Groh and J. Hauffa, "Characterizing social relations via NLP-based sentiment analysis," in *Proceedings of the International Conference on Weblogs and Social Media (ICWSM '11)*, 2011.

[28] M. Miller, C. Sathi, D. Wiesenthal, J. Leskovec, and C. Potts, "Sentiment flow through hyperlink networks," in *Proceedings of the International Conference on Weblogs and Social Media (ICWSM '11)*, 2011.

[29] M. Thelwall, D. Wilkinson, and S. Uppal, "Data mining emotion in social network communication: gender differences in MySpace," *Journal of the American Society for Information Science and Technology*, vol. 61, no. 1, pp. 190–199, 2010.

[30] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-basedmethods for sentiment analysis," *Computational Linguistics*, vol. 37, no. 2, pp. 267–307, 2011.

[31] M. Thelwall, K. Buckley, and G. Paltoglou, "Sentiment in Twitter events," *Journal of the American Society for Information Science and Technology*, vol. 62, no. 2, pp. 406–418, 2011.

[32] M. Thelwall, K. Buckley, G. Paltoglou, M. Skowron, D. Garcia, S. Gobron et al., "Damping sentiment analysis in online communication: discussions, monologs and dialogs," in *Computational Linguistics and Intelligent Text Processing*, pp. 1–12, Springer, 2013.

[33] M. Gamon, "Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis," in *Proceedings of the 20th international conference on Computational Linguistics*, p. 841, 2004.

[34] P. J. Stone, D. C. Dunphy, and M. S. Smith, *The General Inquirer: A Computer Approach to Content Analysis*, MIT Press, 1966.

[35] M. M. Bradley and P. J. Lang, "Affective norms for English words (ANEW): instruction manual and affective ratings," Technical Report C-1, The Center for Research in Psychophysiology, University of Florida, 1999.

[36] S. Baccianella, A. Esuli, and F. Sebastiani, "SentiWordNet 3. 0: an enhanced lexical resource for sentiment analysis and opinion mining," in *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC '10)*, pp. 2200–2204, 2010.

[37] C. Strapparava and A. Valitutti, "WordNet affect: an affective extension of WordNet," in *Proceedings of the International Conference on Language Resources and Evaluation (LREC '04)*, pp. 1083–1086, 2004.

[38] J. W. Pennebaker, M. R. Mehl, and K. G. Niederhoffer, "Psychological aspects of natural language use: our words, our selves," *Annual Review of Psychology*, vol. 54, pp. 547–577, 2003.

[39] A. Neviarouskaya, H. Prendinger, and M. Ishizuka, "SentiFul: a lexicon for sentiment analysis," *IEEE Transactions on Affective Computing*, vol. 2, no. 1, pp. 22–36, 2011.

[40] M. Taboada and J. Grieve, "Analyzing Appraisal automatically," in *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect*, pp. 158–161, AAAI Press, Stanford, Calif, USA, March 2004.

[41] H. Liu, H. Lieberman, and T. Selker, "A model of textual affect sensing using real-world knowledge," in *Proceedings of the 8th International Conference on Intelligent User Interfaces*, pp. 125–132, January 2003.

[42] J. Wiebe and E. Riloff, "Creating subjective and objective sentence classifiers from unannotated texts," in *Computational Linguistics and Intelligent Text Processing*, pp. 486–497, Springer, 2005.

[43] F. Benamara, B. Chardon, Y. Y. Mathieu, and V. Popescu, "Towards context-based subjectivity analysis," in *Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP '11)*, pp. 1180–1188, 2011.

[44] O. Täckström and R. McDonald, "Discovering fine-grained sentiment with latent variable structured prediction models," in *Advances in Information Retrieval*, pp. 368–374, Springer, 2011.

[45] B. Pang and L. Lee, "Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL '05)*, pp. 115–124, June 2005.

[46] A. B. Goldberg and X. Zhu, "Seeing stars when there aren't many stars: graph-based semi-supervised learning for sentiment categorization," in *Proceedings of the 1st Workshop on Graph Based Methods for Natural Language Processing*, pp. 45–52, 2006.

[47] A. Neviarouskaya, H. Prendinger, and M. Ishizuka, "Textual affect sensing for sociable and expressive online communication," in *Affective Computing and Intelligent Interaction*, pp. 218–229, Springer, 2007.

[48] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas, "Sentiment in short strength detection informal text," *Journal of the American Society for Information Science and Technology*, vol. 61, no. 12, pp. 2544–2558, 2010.

[49] R. A. Calvo and S. M. Kim, "Emotions in text: dimensional and categorical models," *Computational Intelligence*, vol. 29, no. 3, pp. 527–543, 2013.

[50] R. A. Calvo and S. D'Mello, "Affect detection: an interdisciplinary review of models, methods, and their applications," *IEEE Transactions on Affective Computing*, vol. 1, no. 1, pp. 18–37, 2010.

[51] P. Ferguson, N. O'Hare, M. Davy et al., "Exploring the use of paragraph-level annotations for sentiment analysis of financial blogs," in *Proceedings of the 1st Workshop on Opinion Mining and Sentiment Analysis*, 2009.

[52] L. Zhang, R. Ghosh, M. Dekhil, M. Hsu, and B. Liu, "Combining lexiconbased and learning-based methods for twitter sentiment analysis," HP Laboratories, Technical Report HPL-2011, vol. 89, 2011.

[53] T. Morrison, *New Paradigm for Robust Combinatorial Optimization: Using Persistence as a Theory of Evidence* , University of Colorado, 2010.

[54] H. Wang, Y. Lu, and C. 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 (KDD '10)*, pp. 783–792, July 2010.

[55] D. Bespalov, Y. Qi, B. Bai, and A. Shokoufandeh, "Sentiment classification with supervised sequence embedding," in *Machine Learning and Knowledge Discovery in Databases*, pp. 159–174, Springer, 2012.

[56] C. Rosé, Y.-C. Wang, Y. Cui et al., "Analyzing collaborative learning processes automatically: exploiting the advances of computational linguistics in computer-supported collaborative learning," *International Journal of Computer-Supported Collaborative Learning*, vol. 3, no. 3, pp. 237–271, 2008.

[57] A. Luneski, E. Konstantinidis, and P. D. Bamidis, "Affective medicine: a review of affective computing efforts in medical informatics," *Methods of Information in Medicine*, vol. 49, no. 3, pp. 207–218, 2010.