Analyzing sentiment system to specify polarity by lexicon-based

Dhafar Hamed Abd¹, Ayad R. Abbas², Ahmed T. Sadiq³

¹,³,⁴Department of Computer Science, University of Technology, Baghdad, Iraq
²Department of Computer Science, Al-Maaref University College, Alanbar, Iraq

ABSTRACT

Currently, sentiment analysis into positive or negative getting more attention from the researchers. With the rapid development of the internet and social media have made people express their views and opinion publicly. Analyzing the sentiment in people views and opinion impact many fields such as services and productions that companies offer. Movie reviewer needs many processing to be prepared to detect emotion, classify them and achieve high accuracy. The difficulties arise due of the structure and grammar of the language and manage the dictionary. We present a system that assigns scores indicating positive or negative opinion to each distinct entity in the text corpus. Propose an innovative formula to compute the polarity score for each word occurring in the text and find it in positive dictionary or negative dictionary we have to remove it from text. After classification, the words are stored in a list that will be used to calculate the accuracy. The results reveal that the system achieved the best result in accuracy of 76.585%.

Keywords:
Dictionary
Lexicon
Movie review
Polarity
Sentiment analyzing
Text analyzing

1. INTRODUCTION

Sentiment analysis also called opinion mining is described as the procedures of mining choose information from a text and realize the attitude which the writer is seeking to articulate through language [1, 2]. Typically, sentiment extractors classify version as either having a positive sentiment or a negative one (or occasionally neutral). A customary function of sentiment test is the programmed ascertainment of whether a web-based review (of a book, movie or consumer product) encompasses a positive or negative evaluation [3]. A more detailed sentiment analysis might encompass distinguishing manifold sentiments embedded within a single text. For example, a well-refined sentiment analysis suite might be able to mine from a review of a restaurant that the specific critic had a favourable opinion regarding the food, but a negative view about the service [4]. In the last two decades, the volume of subjective information obtainable from the Internet has risen considerably. There are now an increasing number of websites where users could generate and share what they want. Especially, social networks are an extremely valued exporter of information which enables users to share their views, thoughts, and sentiments [5].

A few years ago, business entities had no means of finding out what purchasers were contemplating about their product and service offerings, other than through surveys or based on sales figures. With the beginning of the social media, people can access such kind of information, considering users are constantly articulating their views openly, and allowing other views to influence their purchase decisions. Similarly, political organisations can utilise this information for finding out the views of residents on different issues, and also for estimating the intent to vote during an election [6]. Sentiment refers to the views of individuals regarding a topic (for example, a novel product or movie), which can be negative or positive [7].

Journal homepage: http://beei.org
It was around 2007 that analysts and academics realised the significance and worth of social media monitoring, and the importance of sentiment analysis as a tool to achieve it [8]. Sentiment analysis has drawn significant attention of many researchers in the past few years. Researchers have deployed sentiment analysis for evaluating movie reviews, product reviews, poll forecasts, and for commercial intent [9]. Individuals check out the ‘The most online with high reputation’ when selecting a specific product prior to making a specific purchase, instead of collecting feedback from friends/relatives. Organisations can analyse their customer [10].

Many researchers’ studies are based on the emotions expressed in English language as a movie reviewer, but none of them categorise the dictionary into levels and think that the dictionary’ size will not have much impact. The sentiment analysis by movie reviewer concentrates on categorising the sentiment into negative or positive feelings. Reviewers require additional processing to make them ready for identifying and segmenting emotions. This problem stems from the structure and grammar used affecting to the language as well as handling of the dictionary. Sentiment analysis could be executed at various methods, including the following methods.

A number of studied used methods for movie review employ a classifier taken from the machine learning field, which has been made proficient on features of movie review [11]. Machine learning approach can be classified into three categories [12, 13]: unsupervised, supervised and semi-supervised approaches. These are employed to recognise expressions of polarity in text automatically, such as negative and positive. The hybrid approach relies on combining both the machine learning method and the lexicon-based method, which could improve the idea of characterisation execution. Due, Lexicon contains a list of words or phrases and it is the important resource in sentiment analysis [14, 15]. There has been a verity of approaches to constructing lexicon manually or automatically. Lexicon manually is expensive time and does not work with all domain so Lexicon automatically has become a hot research topic because easy to use and work with any domain.

In this paper, to recognise the emotional segmentation of a movie reviewer, an adaptive model is developed including three emotion classes: negative, positive, and neutral, via the lexicon-based method. A sophisticated programming language platform was employed for the implementation of this model as well as assessing the results. In the classification process, the special dictionary is employed as it provides good accuracy. This paper is organised as follows; in section 2, we discuss the related work, while section 3 discusses the modelling. Section 4 illustrates the execution process. Lastly, the outcomes show in section 5, and the conclusion is presented in section 6.

2. RELATED WORKS

In the past few years, numerous researchers have been investigated regarding analyses of people’s thoughts and feelings. Getting the text polarity involves various levels: sentence, full text or even the various entities that were named in the text. Mishra et al. [16] concentrated on analysing the performance pertaining to the ‘Digital India’ campaign. The results of the analysis supported the initiative-20% negative, 50% positive and 30% neutral.

According to Njagi Dennis et al. [17], a model classifier uses sentiment analysis methods for subjectivity detection and to rate the polarity of sentiment sentences. The proposed model starts by deleting the objective sentences. Then, create a lexicon that is used to build a classifier based on features of subjectivity and semantic related to hate speech where this classifier is employed detecting the hate speech. The experiments obtained best results when features of semantic, hate and theme-based were used. Further, both precision and recall are improved when subjective sentences are used.

Based to Asha S. Manek et al. [18], the authors combined the selection method of Gini index based feature with classifier of support vector machine (SVM) to propose sentiment classification model that used for large data set. The conducted outcomes demonstrate improvement is classification performance, reduced error rate and better accuracy. Ankit Sharma et al. [19], this paper used feature-based opinion mining and supervised machine learning to analyse the sentiment of movie reviews. This research extracts nouns, verbs, and adjectives from review and uses them as opinion words to determine the polarity of reviews, where reviews are classified into two different type positive and negative.

In 2019, Rajkumar S. et al. [20] proposed to analyze the sentiments and then classify them by using machine learning algorithms Naïve Bayes (NB) and support vector machine (SVM) with dictionary-based model based one lexicon-based method. NB classifier obtained 98.17% accuracy for camera reviews and SVM achieved 93.54% accuracy for camera reviews. Depending on Shubham Kumar [21], a model used lexicon-based approach to obtain star rating to the reviews. They suggested classifying them into five subclasses, which are excellent, good, neutral, bad and worst based on star-scale method rather than using the classification of positive, negative or neutral. They applied the suggested model with Naïve Bayes and neural network classifiers.
3. THE PROPOSED SYSTEM FOR SENTIMENT EXTRACTION AND CLASSIFICATION

This paper proposes a system that extracts the sentiments from a given text and classifies them. The flowchart in Figure 1 shows the procedure of proposed system which can be explained as the following:

a. Data: Represent the data source and the used data set is corpora and a generated data set.

b. Data Pre-processing: This step consists of tokenization, transforming cases, stemming and filtering stop words and extraction of opinion-oriented words.

c. Shuffling & split documents: This task aims to generate new words by selecting document randomly, shuffling them, and add the document to positive and negative then remove it from source. In this step the document is split into test and train document.

d. Extraction the words: this step represents the classification task where the words are classified into positive or negative. After classification, the words are stored in a list that will be used to calculate the accuracy.

e. Polarity computation: This is the final step of the proposed system of classifying the documents into positive or negative, which is named polarity of the sentiment. This step is repeated until a decision is obtained.

Figure 1. Extract sentiments process

Lexicon contains two types, manually or automatically. Lexicon manually is expensive time and does not work with all domain, so Lexicon automatically has become a hot topic because easy to use and work with any domain. Therefore, to implement our proposed approach, we use a lexicon of over 50,000 movie reviews taken from IMDb is used. Sentiment analysis depends on our ability to identify the terms in a corpus. We denied separate lexicons for each of positive or negative. We selected these dimensions based on our identification of distinct news spheres with distinct standards of opinion and sentiment. Finally, ideally, final step in the process would be a two-way classification task, determining whether the contextual polarity is positive, negative and calculate the accuracy. As shown in Figure 1.

3.1. Dataset collection

In this section, we descry about the datasets that utilized in the performed investigates. A dataset of 50,000 movie reviews taken from IMDb is used by the proposed system. The dataset was compiled by Andrew Maas [22]. It is split evenly with 25,000 reviews intended for training task and 25,000 for testing task. Moreover, each set has 12,500 positive and 12,500 negative reviews. Table 1 illustrates the total number of data sets used in this research.

<table>
<thead>
<tr>
<th>No.</th>
<th>Count</th>
<th>Label</th>
<th>Training &amp; Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12,500</td>
<td>Positive</td>
<td>Training</td>
</tr>
<tr>
<td>1</td>
<td>12,500</td>
<td>Negative</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>12,500</td>
<td>Positive</td>
<td>Testing</td>
</tr>
<tr>
<td>2</td>
<td>12,500</td>
<td>Negative</td>
<td></td>
</tr>
</tbody>
</table>
3.2. Data pre-processing

This step encompasses pre-processing the data so as to make the data all set for analysis [23]. This includes:

a. Tokenisation: In this step, the text of a document is segmented into a sequence of tokens. Normally, the data obtained from the online reviews are associated with noises like HTML tags, URLs, advertisements, scripts and symbols such as hashes, asterisks, etc., which are not important or useful in the classification. Thus, to improve the classifier’s performance, these noise and symbols need to be eliminated so that only the texts are retained.

b. Unwanted characters elimination: This process helps to remove the unwanted characters or string present in the text like non-English characters, English numbers, hashtags, punctuation marks and others. Numerous regular expressions need to be employed to achieve this task, as listed in Table 2.

c. Normalisation: Here, all the characters in a document are converted to either uppercase or lowercase. Most of the reviews use a combination of casing, i.e. uppercase and lowercase characters. In this process, the complete document set is transformed into lowercase. Also, it helps to remove redundant characters from few of the words, like ‘soooon’ is changed to ‘soon’ and ‘gooooood’ is changed to ‘good’ [24].

d. Stop-words removal: This function allows filtering English stop word from a review by eliminating each token, by matching the word with the built-in stop words list. Stop words can be defined as words that are not so important for the opinion or sentence.

e. Stemming: This process involves eliminating the affixes from the word in a bid to make it more concise by using minimum number of words, without affecting the meaning. The put forward system employs Porter stemmer [25].

f. Remove less frequency word: The number of features gets decreased post each pre-processing step, as mentioned in Table 3. Based on the experimental results, it was found that we cannot dodge the pre-processing and data cleaning steps pertaining to English language in a bid to decrease complexity for classifiers, save time and decrease the storage requirements.

Prior to any pre-processing steps, for positive, the number of features is 178,866, while for negative, it is 176,687. However, post regular expression, stemming and stop words, there is a reduction in the number of features to 72,403 for positive while 70,684 for negative. Also, less frequency words can be eliminated to further decrease the numbers of features words.

3.3. Shuffling and split reviewers

This task is performed after the pre-processing is completed. The split process based on n where n is the number data instances used for testing. For example, if there are 1000 data instances 500 positive and 500 negatives and n is given as 200 then the train for positive become 300 and negative 300 so for test positive 200 and negative 200. In our proposal, four different values of (n) are used which are (12,500, 15,000, 17,000, and 20,000).

3.4. Extracting the words

In this section, the system works by dividing the review file into two separate reviews in which the first file contains positive review while the second file contains the negative review. The second step will take into account the ability of distinguishing the positive and negative dictionary files which were separated earlier and calculate the accuracy through a comparison with the original review.
Let $D$ collection of reviewer documents $D = \{X_1, X_2, X_3, \ldots, X_n\}$ and $L$ label of each documents $L = \{\text{positive, negative}\}$, each document $X_n \in L$. Assume $P$ for positive dictionary and $N$ for negative dictionary for building $P$ and $N$ using formula below.

$$P = \bigcup X_n \mid X_n \in L_{\text{positive}}$$

$$N = \bigcup X_n \mid X_n \in L_{\text{negative}}$$

(1)

### 3.5. Polarity computation

We propose an innovative formula to compute the polarity score for each word occurring in the text and find it in positive dictionary or negative dictionary we have to remove it from text. The computed score will range from 0 to $n$. A less words in review text it will be that polarity of review text. Negative value represents a less words their negative sentiment and a less positive value represents a positive sentiment.

In the proposed algorithm, the following notations or parameters are introduced for explanation in Table 4.

<table>
<thead>
<tr>
<th>Parameter of our model</th>
<th>Explain</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P$</td>
<td>Positive words in dictionary</td>
</tr>
<tr>
<td>$N$</td>
<td>Negative words in dictionary</td>
</tr>
<tr>
<td>$R$</td>
<td>Reviewer text</td>
</tr>
<tr>
<td>$W$</td>
<td>Word that appear in document</td>
</tr>
<tr>
<td>$ocr(W, P)$</td>
<td>The number of positive words that not be contain $w$</td>
</tr>
<tr>
<td>$ocr(W, N)$</td>
<td>The number of negative words that not be contain $w$</td>
</tr>
</tbody>
</table>

The polarity of review text is calculated by the proposed formula:

$$ocr(W, P) = \{w \mid w \in R \text{ and } w \notin P\}$$

$$ocr(W, N) = \{w \mid w \in R \text{ and } w \notin N\}$$

(2)

After we have calculated the number of words not found in both $P$ and $N$ dictionary, then we use the formula below to calculate which less words in order for the target is,

$$Polarity = \begin{cases} 
  \text{positive} & \text{if } ocr(w, N) > ocr(w, P) \\
  \text{negative} & \text{if } ocr(w, N) < ocr(w, P) \\
  \text{neutral} & \text{if } ocr(w, N) = ocr(w, P)
\end{cases}$$

(3)

If the $ocr(w, N)$ of the negative upper than the $ocr(w, N)$ of the positive, it means the accuracy is positive and also on the contrary, if the $ocr(w, N)$ is equal to the $ocr(w, N)$, it means neutral.

Our simulation procedure used Python programming language where Python is widely used high-level programming language [26]. Pre-processing task is performed by using NLTK. It is most popular library for natural language processing. The result build on which situation a classifier got a right prediction or not classifications, which can be measured based on the equation:

$$\frac{T+N}{R}$$

(4)

where $T$ is the true (positive or negative), $N$ is the neutral and $R$ is all review text of (positive or negative).

### 4. RESULTS AND DISCUSSION

This section presents the number of documents and words for the training and testing review text in our experimental study as shown in Table 5. In order to obtain satisfied results, we need to divide the datasets into 4 levels for the positive and negative words. The comparison is implemented based on target value decision on the entire of text mining datasets, which are used to evaluate the total number of levels (level 1, level 2, and level 3, level 4). As shown in Table 6, level 1 received the highest number of positive, neutral and error 5755, 3775, and 2970, respectively, while, negative, neutral, and error obtained 5810, 3688, and 2997 respectively.
To explore how much the various polarity contribute to the performance of the polarity classifier, we perform four experiments. In each experiment, a different set of polarity words used, and the polarity classifier is evaluated. Table 6, 7 lists of each experiment. We found a dictionary at the second level with an accuracy of 76.585% where 30,000 documents were taken to positive and negative, note that there was no strong effect between the four levels or no significant difference. The ratios were almost the same.

Tables 6 and 7 show the train documents, test documents and concentrated the results mostly on the positive words and negative. Table 7 shows the distribution percentages for the negative/positive and neutral. The results showed that the proposed method ability to detect the positive/negative and neutral words and achieved lower classification error rates

Table 5. Building dictionary within documents and words

<table>
<thead>
<tr>
<th>No</th>
<th>Building dictionary</th>
<th>Test documents number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Positive Words</td>
<td>Negative Words</td>
</tr>
<tr>
<td>L1</td>
<td>12,500</td>
<td>18,778</td>
</tr>
<tr>
<td>L2</td>
<td>15,000</td>
<td>20,773</td>
</tr>
<tr>
<td>L3</td>
<td>17,000</td>
<td>22,268</td>
</tr>
<tr>
<td>L4</td>
<td>20,000</td>
<td>24,359</td>
</tr>
</tbody>
</table>

Table 6. Target value decision results

<table>
<thead>
<tr>
<th>No</th>
<th>Positive</th>
<th>Neutral</th>
<th>Error</th>
<th>Negative</th>
<th>Neutral</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1</td>
<td>5755</td>
<td>3775</td>
<td>2970</td>
<td>5810</td>
<td>3688</td>
<td>2997</td>
</tr>
<tr>
<td>L2</td>
<td>4390</td>
<td>3280</td>
<td>2530</td>
<td>4501</td>
<td>3146</td>
<td>2353</td>
</tr>
<tr>
<td>L3</td>
<td>3284</td>
<td>2706</td>
<td>2010</td>
<td>3516</td>
<td>2624</td>
<td>1860</td>
</tr>
<tr>
<td>L4</td>
<td>1947</td>
<td>1758</td>
<td>1295</td>
<td>2176</td>
<td>1719</td>
<td>1105</td>
</tr>
</tbody>
</table>

5. CONCLUSION AND FUTURE WORK

This paper presented a system that detects the polarity by analyzing sentiment, the proposal on lexicon with dictionary methods. This paper aimed on the lexicon-based approach to sentiment analyzing since it is one of the most widely studied approaches. We present a new approach to phrase-level sentiment analysis that first determines whether is positive or negative. With this approach, we are able to automatically identify the contextual polarity for a large subset of sentiment, achieving results that are significantly better. For this effect, we built a sentiment lexicon of about 25,000, 30,000, 34,000, and 40,000 review terms and built a SA based on dictionary. The proposed approach showed great results, in terms of prediction accuracy. The results maintain its accuracy even if the size of the dictionary is changed does not affect on result. Our suggestion for the future work, the dictionary idea can apply very quickly and its speed and the machine learning to get the high accuracy

REFERENCES


**BIographies of Authors**

**Dhafar Hamed Abd** received his B.Sc. Degree in Information System from the College of Computer, University of Anbar, 2008, Iraq. He gained his Master in Information System from Osmania Universiti 2014, India. He is currently PhD candidate at Technology University. His research interests include data science, artificial intelligence, machine learning, Natural Language Processing and advanced algorithm development.

**Ayad R. Abbas** is a Principal Lecturer in Applied Computing at the faculty of Computer Science and Environment at Technology University. He is the Head of Computer Science for the Faculty. Dr. Abbas has extensive research interests covering a wide variety of interdisciplinary perspectives concerning the theory and practice of Applied Computing in Text Mining. He has published well over 40 peer reviewed scientific publications and 3 book chapters, in multidisciplinary research areas including: Technology Enhanced Learning and Applied Artificial Intelligence.

**Ahmed Tariq Saadeq** is a Reader (professor) in Artificial Intelligence and Data Mining and he is the deputy head of the Applied Computing Research Group at the faculty of Technology. He completed his PhD study at The University of Technology, Iraq. He has published numerous referred research papers in conferences and Journal in the research areas of AI, Data Mining and Text Mining. He is a PhD supervisor and an external examiner for research degrees including PhD and MPhil.

---

*Analyzing sentiment system to specify polarity by lexicon-based (Dhafar Hamed Abd)*