Analysis on Polarity Classification of Product Review Data Using Vader Rule Based Approach

Hlaing Myo Zaw,
University of Technology-Yatanarpon Cyber City
hlaingmyozaw@utycc.edu.mm

Myat Su Wai
University of Technology-Yatanarpon Cyber City
myatsuwai@ucsm.edu.mm

Abstract

Sentiment analysis has become popular in linguistic researches. Opinions for such analysis are drawn from many forms of freely available online/ electronic sources, such as websites, blogs, news re-ports and product reviews. In this paper, a given input sentence is classified as either a positive or negative depending upon the opinion's polarity score by using Vader rule-based approach. Some occasions, the opinion does not say anything about the topic being talked about; such neutral opinions can be considered as objective opinions and ignore them. Moreover, to show the domain independency of our previous work, we set our experiment by combining all datasets into one, input to the system and analyzed them. The results of this domain independency evaluation are calculated in term of precision, recall and f1-score. Finally, the accuracy of the polarity classification is described and analyzed.

***Keywords**: Sentiment Analysis, Product Reviews, polarity classification.*

1. Introduction

Opinions are subjective expressions of human thoughts, emotions and feelings. The research area of analyzing opinions contained in texts is popularly known as opinion mining or sentiment analysis. To fulfil this aim, both opinion targets and opinion words must be detected. First, however, it is necessary to extract and construct an opinion target list and an opinion word lexicon, both of which can provide prior knowledge that is useful for fine-grained opinion mining and both of which are the focus of this paper.

A given opinion can be classified as either a positive or negative or objective one, depending upon the opinion's polarity towards or against the theme of the topic being talked about. Some occasions, the opinion does not say anything about the topic being talked about; such neutral opinions can be considered as objective opinions. This whole procedure is known and subjectivity classification as in [10] and [12].

The primary resource required for classifying an opinion based on the above de-scribed categories using a supervised method in a given language is, the lexicon type repository called Sentiment lexicon. Sentiment lexicon usually contains a special set of words for the language with polarity scores either positive or negative. The polarity score is a scale used to determine the sense of the word that is present in the opinion.

Hence, an opinion can be categorized into positive, negative or objective by combining the polarity scores of the sentiment words in that opinion. Social media texts are classified into positive, negative and neutral score in the original Vader sentiment analysis approach. In this paper, product reviews are classified into positive and negative based on the score obtained from Vader approach. This paper contributed the following point:

- Testing domain independency of this system by combining all different datasets into one file.
- Calculating positive, negative and neutral score upon the input product review data using Vader sentiment analysis method.
- Classification into positive, negative opinion based on the score/

The rest of this paper is organized as follow. Section 2 describes about the related works. Section 3 presents the proposed method of this work. Section 4 explains about the experimental results. Section 5 concludes this paper.

2. Related Work

Several researches have been investigated to automatically generate resources for a new language starting from lexical resources already available for the English language.

[1] presented VADER, a simple rule-based model for general sentiment analysis, and compare its effectiveness to eleven typical state-of-practice benchmarks including LIWC, ANEW, the General Inquirer, SentiWordNet, and machine learning oriented techniques relying on Naive Bayes, Maximum Entropy, and Support Vector Machine (SVM) algorithms.

[2] reported their works on aspect term extraction and sentiment classification with respect to our participation in the SemEval-2014 shared task. The aspect term extraction method is based on supervised learning algorithm, where we use different classifiers, and finally combine their outputs using a majority voting technique.

[3] made a survey focused on aspect-level sentiment analysis, where the goal is to find and aggregate sentiment on entities mentioned within documents or aspects of them. Aspect-level sentiment analysis yields very fine-grained sentiment information which can be useful for applications in various domains. Current solutions are categorized based on whether they provide a method for aspect detection, sentiment analysis, or both. Furthermore, a breakdown based on the type of algorithm used is provided.

[4] have studied opinion mining at the document, sentence and aspect levels. Aspect-level is often desired

in practical applications as it provides the detailed opinions or sentiments about different aspects of entities and entities themselves, which are usually required for action. Aspect extraction and entity extraction are thus two main tasks of aspect-based opinion mining.

[5] aimed to find and to summarize all the user reviews of a product. This summarization task is different from traditional text summarization because the authors only find the features of the product on which the customers have expressed their opinions and whether the opinions are positive or negative. The authors do not summarize the reviews by selecting a subset or rewrite some of the original sentences from the reviews to capture the main points as in the classic text summarization.

[8] discussed existing techniques and approaches for feature extraction in sentiment analysis and opinion mining. In this review, they have adopted a systematic literature review process to identify areas well focused by researchers, least addressed areas are also highlighted giving an opportunity to researchers for further work.

[9] proposed a method to select an effective set of rules. This is the first work that selects rules automatically. This approach is highly desirable in practice because it is unsupervised and domain independent. However, the rules need to be carefully selected and tuned manually so as not to produce too many errors.

[10] has tried to address this issue in the mobile app domain. Along with aspect and opinion extraction this work has also categorized the extracted aspects according to their importance. This can help developers to focus their time and energy at the right place.

[11] introduced their research plan to use neural networks on user-generated travel reviews to generate summaries that take into account shifting opinions over time. They outline future directions in summarization to address all of these issues.

[13] a framework is proposed to extract product features from Chinese consumer reviews and construct product feature structure tree. Through three filtering algorithms and two-stage optimizing word segmentation process, phrases are identified from consumer reviews. And the expanded rule template, which consists of elements: phrase, POS, dependency relation, governing word, and opinion, is instructed to train the model of conditional random filed (CRF).

Our methods is different from the existing state-of-the-art methods and original Vader sentiment analysis method because original Vader is based on twitter data and calculating scores of input data. Our method is based on product review data and calculated scores and classified positive or negative based on scores

3. Proposed Method

In the previous work, [6] we performed the opinion lexicon expansion and features extraction tasks simultaneously using the modified algorithm and dependency elations.

Firstly, the system did word tokenization, part-of speech tagging and syntax or dependency analysis as preprocessing of the input sentences. StandfordfordCoreNLP dependency parser is used to identify dependency relations. [12] To start the extraction process with algorithm, a seed opinion lexicon, a list of general words, review data and extraction rules are input to the proposed algorithm. The extraction process uses a rule-based approach using the relations defined in above. The system assumed opinion words to be adjectives, adverbs and verbs in some cases. And product features are nouns or noun phrases and also verbs in some cases.

The algorithm ends until no more new opinion words and features can be found. Even if the seed opinion lexicon is small, features can be extracted and at the same time the opinion lexicon is also expanded. Incorrect features are removed by using general word set obtained from WordNet and NLTK [11] and [12].

After extraction step, the extracted opinion words are calculated polarity scores and classified positive or negative. In this step, we use Vader sentiment analysis method for classifying the input sentence as positive, negative or neutral.

[7] The system also considers whether there are negations/contrary words associated with the opinion words. The negation/contrary word set consists of not, n't,'t, however, but, despite, though, except, although, oddly, and aside. If the sentences contain these negation words are opposite in their polarity orientation.

3.1. Quantifying the Emotion of a Word

VADER sentiment analysis relies on a dictionary which maps lexical features to emotion intensities called sentiment scores. The sentiment score of a text can be obtained by summing up the intensity of each word in the text.

In a typical sentence, we can find not only words, but also emoticons like ":-)", acronyms like "LOL", and slang like "meh". The cool thing about VADER sentiment analysis is that these colloquialisms get mapped to intensity values as well [1] and [7].

Emotion intensity or sentiment score is measured on a scale from -4 to +4, where -4 is the most negative and +4 are the most positive. The midpoint 0 represents a neutral sentiment. Sample entries in the dictionary are "horrible" and "okay," which get mapped to -2.5 and 0.9, respectively. In addition, the emoticons "/-:" and "0: -3" get mapped to -1.3 and 1.5.

Some words might not seem very negative to you, but they might be to me. To counter this, the creators of VADER sentiment analysis enlisted not just one, but a number of human ratters and averaged their ratings for each word. This relies on the concept of the wisdom of the crowd: collective opinion is oftentimes more trustworthy than individual opinion.[1] [7].

3.2. Quantifying the Emotion of a Sentence

VADER sentiment analysis returns a sentiment score in the range -1 to 1, from most negative to most positive.

Cautious readers would probably notice that there is a contradiction: individual words have a sentiment score between -4 to 4, but the returned sentiment score of a sentence is between -1 to 1 [1] [7].

The sentiment score of a sentence is the sum of the sentiment score of each sentiment-bearing word. However, we apply normalization to the total to map it to a value between -1 to 1. The normalization used in this system is

$$x / (\sqrt{x2+\alpha})$$

where x is the sum of the sentiment scores of the constituent words of the sentence and α is a normalization parameter that we set to 15. Thus, VADER sentiment analysis works best on short documents, like sentences and sentences, not on large documents [1] and [7].

3.3. Five Simple Rules

Lexical features aren't the only things in the sentence which affect the sentiment. There are other contextual elements, like punctuation, capitalization, and modifiers which also impart emotion. VADER sentiment analysis takes these into account by considering five simple heuristics. The effect of these heuristics is, again, quantified using human ratters.

The first rule is punctuation. Compare "I like it." and "I like it!!!" It's not really hard to argue that the second sentence has more intense emotion than the first, and therefore must have a higher sentiment score.

This method takes this into account by amplifying the sentiment score of the sentence proportional to the number of exclamation points and question marks ending the sentence. It first calculates the sentiment score of the sentence. If the score is positive, the system adds a certain empirically-obtained quantity for every exclamation point (0.292) and question mark (0.18). If the score is negative, the system subtracts 0.292 [1].

The second rule is capitalization. "AMAZING performance." is definitely more intense than "amazing performance." The system takes this into account by incrementing or decrementing the sentiment score of the word by 0.733, depending on whether the word is positive or negative, respectively [1].

The third rule is the use of degree modifiers. The effect of the modifier in the first sentence is to increase the intensity of cute, while in the second sentence, it is to decrease the intensity. The system maintains a booster dictionary which contains a set of boosters and dampeners.

The effect of the degree modifier also depends on its distance to the word it's modifying. Farther words have a relatively smaller intensifying effect on the base word. One modifier beside the base word adds or subtracts 0.293 to the sentiment score of the sentence, depending on whether the base word is positive or not. A second modifier from the base word adds/subtract 95% of 0.293, and a third adds/subtracts 90% [1] [7].

The fourth rule is the shift in polarity due to "but". Oftentimes, "but" connects two clauses with contrasting sentiments. The dominant sentiment, however, is the latter one. The system implements a "but" checker. Basically, all sentiment-bearing words before the "but" have their valence reduced to 50% of their values, while those after the "but" increase to 150% of their values [6].

Table 1 Negation Words

cannot	can't	couldn't	didn't
doesn't	don't	aren't	hadn't
hasn't	haven't	isn't	mightn't
mustn't	needn't	never	none
nor	not	nothing	nowhere
oughtn't	shan't	shouldn't	wasn't
weren't	without	wouldn't	won't
rarely	seldom	despite	neither

The fifth rule is examining the tri-gram before a sentiment-laden lexical feature to catch polarity negation. Here, a tri-gram refers to a set of three lexical features. The system maintains a list of negation words. Negation is captured by multiplying the sentiment score of the sentiment-laden lexical feature by an empirically-determined value -0.74. Table 1 shows some negation word considered in this system.

Finally, the system classifies the input sentence into positive or negative based on the scores obtained from the above steps. If the score is greater than 0, the system decides the input sentence as positive. Otherwise, it is negative.

4. Experimental and Discussion

For experiment, we use core i7 processor with 2.20 GHz speed (2 gen), 8GB RAM with 1333 MHz speed and 64-bit Ubuntu OS, and, the proposed system is implemented with python programming language (PyCharm IDE for python).

In this work, 10 reviews datasets are chosen to test the proposed system for experiment. To experiment domain independency, all of these datasets are combined into one dataset.

The performance of the proposed system is evaluated in opinion words extraction and features extraction and polarity classification. In the previous work, the performance evaluation of feature extraction, and opinion extraction are evaluated in all datasets.

In this work, we use another evaluation style to test domain independency of this system. All datasets are combined into one file and tested in the process of feature extraction and opinion extraction. We applied precision, recall and f-measure to measure the performance of the proposed system.

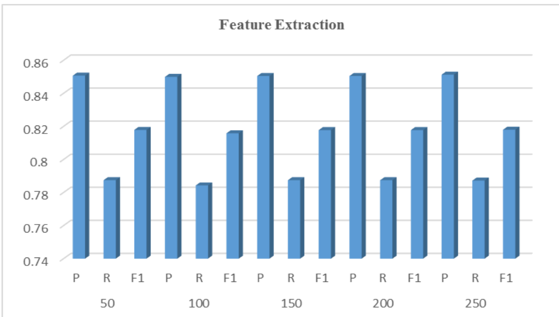

Figure 1. Experimental Results of Feature Extraction on all datasets

Figure 1 describes performance analysis of feature extraction. In figure 1, X axis shows seed numbers used in this system and Y axis shows precision, recall, and f1 score of feature extraction result of all datasets. The results are stable although the different seed numbers and lexicons are changed. The proposed approach achieves 85% in precision, 78% in recall in and 82% in f1-score respectively.

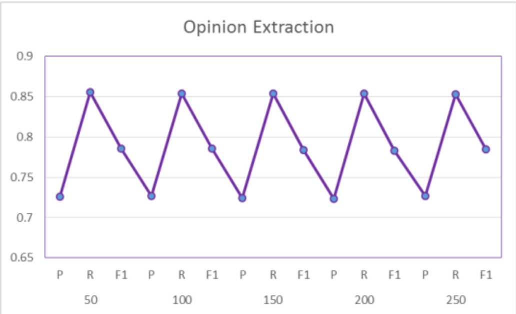

Figure 2. Experimental Results of Opinion Extraction on all datasets

In opinion lexicon expansion, the proposed approach is evaluated on newly extracted words without considering the seeds. Different seeds number such as 50, 100, 150, 200 and 250 words from Bing Liu's lexicon. This is because we intended to consider this system to evaluate different number of seed words.

In figure 2, X axis shows seed numbers used in this system and Y axis shows precision, recall, and f1 score of opinion word expansion of all datasets. According to the results, the proposed approach achieves 85% in precision, 78% in recall in and 82% in f1-score respectively.

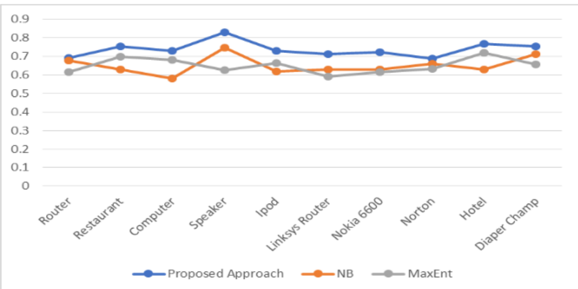

Figure 3. Comparative Results of Polarity Classification

To experiment the Naïve Bayes (NB) and Maximum Entropy (MaxEnt), opinion words from each dataset are used as training data and input sentences of these datasets are used as testing data. The experimental results of polarity classification in all datasets are compared and analyzed with the proposed approach.

In figure 3, X axis shows different datasets used in this system and Y axis shows accuracy score of polarity orientation of all datasets. According to the comparative results, the proposed approach achieves highest accuracy 83% in speaker dataset and gets better performance than two other approaches. The system gets better performance than the two other supervised learning methods, Naïve Bayes and Maximum Entropy because NB and Naximum Entropy methods can only work well in simple and short sentences. They cannot classify correctly if the sentences contain negation words and intensifier words. The proposed approach uses vader lexicon and booster word, intensifier words (degree modifier) and negation words and their corresponding score. So, the sentences with negation words and degree modifier can handle effectively.

5. Conclusion

Opinions are important because they are key influencers of our behaviors. For this reason, when we need to make a decision, we often seek out the opinions of others. This is true not only for individuals but also for organizations. In this work, sentiment classification is done by using Vader sentiment analysis method for the product reviews. This method is good because it considered negation, degree modifier, contrary words, and emotion. In the previous work, Features and opinions words are extracted simultaneously by using the proposed algorithm. The results of the previous work are also described because domain independencies of our work are evaluated in this work. According to experimental results, the proposed system works well in feature extraction, opinion extraction and polarity classification. As the future works, when the new version of StanfordCoreNLP is released, more dependency relation between words will be handled and improve the performance of this work.

References

[1] C. J. Hutto and E. Gilbert, "VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text," May 2014, doi: null.

[2] D. K. Gupta and A. Ekbal, "IITP: Supervised Machine Learning for Aspect based Sentiment Analysis," in Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland, 2014, pp. 319–323, doi: 10.3115/v1/S14-2053.

[3] K. Schouten and F. Frasincar, "Survey on Aspect-Level Sentiment Analysis," IEEE Trans. Knowl. Data Eng., vol. 28, no. 3, pp. 813–830, Mar. 2016, doi: 10.1109/TKDE.2015.2485209.

[4] L. Zhang and B. Liu, "Aspect and Entity Extraction for Opinion Mining," in Data Mining and Knowledge Discovery for Big Data, vol. 1, W. W. Chu, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 1–40.

[5] M. Hu and B. Liu, "Mining and Summarizing Customer Reviews," p. 10.

[6] M. S. Wai and S. S. Aung, "Simultaneous opinion lexicon expansion and product feature extraction," in 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS), May 2017, pp. 107–112, doi: 10.1109/ICIS.2017.7959978.

[7] M. S. Wai and S. S. Aung, "An Effective Approach to Automatic Feature Based Opinion Lexicon Expansion," Int.

J. Res. Adv. Comput. Sci. Eng. ISSN 2208-2107, vol. 3, no. 6, Art. no. 6, Jun. 2017.
[8] M. Z. Asghar, A. Khan, S. Ahmad, and F. M. Kundi, “A Review of Feature Extraction in Sentiment Analysis,” p. 6, 2014.
[9] Q. Liu, Z. Gao, B. Liu, and Y. Zhang, “Automated Rule Selection for Aspect Extraction in Opinion Mining,” p. 7.
[10] S. Dey, “Aspect Extraction and Sentiment Classifica-tion of Mobile Apps using App-Store Reviews,” p. 12.
[11] S. Pecar, “Towards Opinion Summarization of Customer Reviews,” in Proceedings of ACL 2018, Student Research Workshop, Melbourne, Australia, 2018, pp. 1–8, doi: 10.18653/v1/P18-3001.
[12] S. S. Aung and M. S. Wai, “Domain Independent Feature Extraction using Rule Based Approach,” Adv. Sci. Technol. Eng. Syst. J., vol. 3, no. 1, pp. 218–224, Jan. 2018, doi: 10.25046/aj030126.
[13] X. Xu, J. Lin, Y. Xiao, and J. Yu, “An Approach to Extract Product Features from Chinese Consumer Reviews and Establish Product Feature Structure Tree,” p. 42.