ISSN 2395-8618

Simple and Effective Feature Based Sentiment Analysis on Product Reviews using Domain Specific Sentiment Scores

Nachiappan Chockalingam

***Abstract*—Reviews are a valuable resource. Conclusions drawn on analysis of reviews are of great help in improving the product, as far as the manufacturer is concerned, or with predicting sales figures, as far as the retailer is involved. However, employing human labor to go through all the reviews manually would be a time consuming and expensive process. This paper outlines a novel technique to extract features from a product's reviews along with the corresponding sentiment expressed, using POS tagging and Dependency Parsing in conjunction. The use of these of these allows both the context and the parts of speech of a word to be employed in feature and corresponding opinion word detection. The opinion word is given a sentiment polarity determined from a training set of positive and negative reviews.The method described in this paper is for large data sets, and requires no domain specific data for feature extraction.**

***Index Terms*—Review mining, dataset, sentiment analysis, features, parts of speech tagging, opinion word, dependency parsing.**

I. Introduction

REVIEWS are a set of sentiments expressed over a very short period of time about a product and it's features. The number of reviews and reviewers are only increasing by the day; a trend that shows no sign of abating. Hence, the idea of review analysis to tap into this goldmine of freely available data is alluring.

Numerous systems talk about sentiment analysis to gain the 'average' response for a product [1], [2]. This one dimensional take on the issue ignores the potential for a multi-faceted approach where even individual features of a product can be extracted and analysed. After all, why not use the average star rating? Why even enter text analysis if not to extract 'more' information about/from reviews.

The aim of the proposed system is to extract features from reviews using a series of techniques. Evaluation formulas of precision and recall allow for classification of problems of feature extraction. These being, either find a lot of features but accept a low precision score since there would be a number of unwanted features included in the feature list, or gain in precision by applying additional filtering to the feature list while contending with the possibility of loss of genuine features.

Following feature extraction, polarity classification is done. This step involves assigning scores to opinion words. The opinion words are associated with a feature, and hence the score for the opinion word is linked with that feature. This system works best when a large number of reviews are input (since each feature needs sufficient opinion words describing it).

A. Problem Statement

Given reviews of a particular product, the aim is to summarise the reviews by picking out features and their corresponding opinion words with polarity scores [3].

The Screen is bright and clear-Using this example sentence, the problem statement is explained in steps.

1. Extract all features from given reviews: As there is no previous data about the features to look for, they have to be generated on the go, from the data. Eg: Screen
2. Generate opinion word: The opinion words are extracted, again in the absence of a specific domain. Eg: Bright and Clear
3. Generate Opinion Scores: While feature extraction is not domain specific, the opinion word scores are machine learnt, and hence can be domain specific. Negation and conjunction must be handled. Eg: Bright (Positive), Clear (Positive)
4. Put all the feature analysis together to generate a feature score that is more accurate (hence the large dataset).

B. Literature Survey

Many systems have been proposed for the analysis of reviews and the work on this domain has been on-going for close to two decades. Growth in required and related fields such as e-commerce, computational power, machine learning, and most importantly, Text Analytics has allowed crossing of barriers previously applied on researchers working in this field.

The typical Text Analysis approach uses Cleaning (pre-processing), Analysis and result generation. However, within each broad step, techniques used differ between each sentiment analysis system.

There are different types of sentiment analysis including sentence Level, document Level, aspect-based mining, etc. [4]. All of these are dependent on the domain and aim.

Manuscript received on December 28, 2017, accepted for publication on March 15, 2018, published on June 30, 2018.

The author is with the Department of Computer Science and Engineering, College of Engineering, Guindy, Chennai, Tamil Nadu, 600025, India (e-mail: nach729@hotmail.com).

ISSN 2395-8618

One of the earliest sentiment mining methods included the classification of sentences into positive and negative [5], [6] groups. Further work involved a comprehensive entry into sentence and document level sentiment analysis. Document level analysis is used in a similar case as sentences level analysis since "sentences are just short documents" [7]. Aspect based sentiment analysis is a reference to the level of rating. It allows for identification of features and generating their polarity from the reviews, as opposed to polarity classification of reviews as a whole [4]. Aspect based sentiment analysis uses sentence/document analysis combined with aspect level rating.

Bag of words model is a famous Text Mining [8] approach where the un-needed parts of a text are discarded in favour of keeping ones that are necessary. Many older systems relied on the use of stop words removal as a method to extract desired data. Instead of that approach, the ability to POS tag a sentence coupled with tuple analysis allows for extraction of desired data directly [3], instead of discarding unwanted text. The use of dependency tagging helps maintain context of the word [9].

With regards to polarity determination, Ohana et all [6] used Senti-Word Net to get the word sentiments for identified opinion words. Synsets (sentiment scores) for a particular word were taken and averaged to generate it's polarity. But this lacks domain specific identity (Section 2.1) that provides an authentic score for any specific domain. This is its greatest pitfall.

II. Proposed System

The system proposed uses Parts of Speech Tagging (POS) to parse sentences into constituent elements while Dependency Parsing is used to determine the relationships between words. A rule based analysis can be applied, using which the features and opinion words are extracted. Finally, a sum of all the analysis gives us the perception of each feature.

A. Sentiment Scores

In the related work section, there are issues with determining of sentiment scores for other approaches using pre-determined or previously calculated sentiment scores [6] for opinion words. So, for example, the sentiment score for the opinion word 'sad' is applied across electronics reviews, as well as movie reviews. This leads to inaccurate results since the same opinion word does not correspond to the same sentiment across domains. While the word 'bad' might be acceptable as a universal negative sentiment modifier, many other words do not carry a universal sentiment.

A simple machine learning system with domain specific dataset is used in this system, where the input dataset is of the same domain as the reviews to be analysed. To begin, two datasets- positive and negative are input. Next, each review undergoes pre-processing (Section 2.3) and analysis (Section 2.4) steps outlined later in this paper. During tuple analysis sentences are parsed into nouns and adjectives(opinion words).

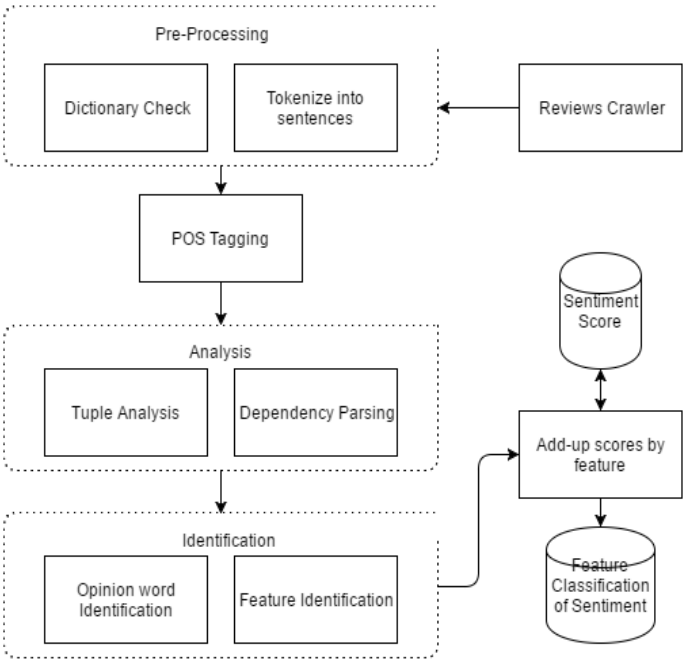

Fig. 1. Method overview.

The fact that we know the polarity of the review input allows us to classify the opinion words into two classes- positive and negative. Each opinion word has a negative and positive counter and every time an opinion word is identified, the counter is iterated for either the positive or negative respectively as found [2].

The training corpus for negative and positive reviews is from the work by Ganapathibhotla and Bing Liu [10].

$$PositiveScore = \frac{PositiveCounter}{PositiveCounter + NegativeCounter}$$

The opinion word score is positive biased. That is to say, all scores are a continuous from 0 to 1. 0 being the most negative and 1 being the most positive. When a word makes no appearance in negative or positive datasets as an opinion word, then the word (as per the formula) will be assigned a score of 1 or 0 respectively. A score of 0.5 represents a neutral sentiment.

B. Pre-Processing

In this stage, the input reviews need to be brought to a format convenient for analysis. Reviews are pushed through a dictionary correction module, parsed into sentences and sent into the analysis system one by one.

C. Analysis

The analysis stage is split into two- the tuple analysis and the dependency parsing. The input to both stages is done after the sentence is POS tagged.

ISSN 2395-8618

1) Tuple Analysis: Tuple analysis involves taking a sentence and turning it into a tuple. A tuple is a stripped version of a sentence, in that it contains only essential parts required for analysis. For example, the Person, Nouns, Adjectives, time, etc found in a sentence are stored in a tuple, and hence it represents what is relevant (to the analysis) in that sentence. For this system, nouns and adjectives are extracted in the absence of Domain Knowledge [3]. The nouns are henceforth referred to as 'potential features', while the adjectives are 'potential opinion words'. Hence a sentence is reduced to:

```
<Potential Features;
Potential Opinion Words>
```

A number of important relationships between words that affect the identification of features and their corresponding sentiment scores remain unknown such as which opinion word corresponds to which feature, conjuntion and negation in the sentence, etc.

2) Dependency Parsing: The dependency parser is effective in taking the POS tagged sentences and obtain the relationship between words. This section can also be called relation extraction, [3] as stated by Mukherjee et al:

Let Dependency Relation be the list of significant relations. We call any dependency relation significant, if

- It involves any subject, object or agent like nounSubject, dobject, agent etc
- It involves any modifier like adverbModifier, adjectiveModifier etc
- It involves negation
- It involves any adjectival or clausal component like clauseModifier

Dependency parsing gives us the relationship between words that can be exploited to generate features and their corresponding sentiment scores from potential features and sentiment scores respectively. Dependency parsing prunes the list of potential features and links them with the specific opinion word associated. The negation (explicitly) and conjunction (implicitly) handling is also done in this stage.

Example 1: **The Phone came yesterday and the display is not very good.**

After POS tagging we get: The(determinant) phone(noun) came(verb) yesterday(noun) and(conjunction) the(determinant) display(noun) is(verb) not(adverb) good(adjective).

Dependency Parsing using Stanford Parser(only relevant tags):

nounSubject(has, phone)
negative(good, not)
adjectiveModifier(display,great)
RelativeClauseModifier(performance, satisfactory)

The negation handling is done using the following algorithm:

```
if neg
    score=(1-score_of_opinion_word)
```

3) Example 2:: **The Phone has a great display and the performance is satisfactory**

After POS tagging we get: The(determinant) phone(noun) has(verb) a(determinant) great (adjective) display(noun) and(conjunction) the(determinant) performance(noun) is(verb) satisfactory(adjective).

Tuple Analysis: phone, display, performance; great, satisfactory

Dependency Parsing (only relevant tags):

nounSubject(has, phone)
adjectiveModifier(display,great)
RelativeClauseModifier(performance, satisfactory)

A combination of both tuple analysis and dependency parsing gives us the desired result. While the dependency parser identifies that the opinion word 'great' relates to display and that satisfactory relates to performance, it also identifies nounSubject(has, phone) which is irrelevant but is within potential relation tags. This irrelevant part is revealed using the POS tagger and pruned, as the relationship does not have a potential opinion and potential feature word. Hence, we get 2 relations: (display, great) and (performance, satisfactory).

D. Issues

This review based analysis technique has the potential to give a reasonably decent accuracy score, but will have low recall score because many sentences have their features mentioned implicitly as opposed to explicitly. Eg :-

It is bright

The system cannot recognize the reference to the screen, and hence will fail in such conditions. Similarly opinions that are not expressly stated will be overlooked. Eg :-

The phone held its own

While the phone is to get a positive polarity associated with it from this review, as the system does not recognize phrases- there is a failure in analysis. Phrase substitution [7] requires separate study to detail an effective method to determine polarity of phrases. This system therefore ignores phrase analysis.

Also, not all noun-adjective pairs are feature-opinion relations. This is the failure of the system.

E. Design Choices

N-gram extraction is a technique often used in review analysis. Since the reviews are so focused (single product) and the products used in analysis have but few features (unlike cars for example), the idea of using n-gram was dropped in favour of unigram extraction. This decision was made at evaluation because of duplicity of feature results like- display quality and screen clarity being classified as 2 different aspects. In the absence of an ontology (Section 4.1), the problem gets further compounded.

The system will recognize a feature only if a sentiment is associated with it. So the sentence: **"The Camera has a strap"** will not have strap recognized as a feature. The system

ISSN 2395-8618

TABLE I
EVALUATION OF FEATURE IDENTIFICATION.

Review Domain	Precision	Recall	F-measure	Comparable System
MP3 Player	0.71	0.82	0.87	0.64
Camera	0.60	0.83	0.697	0.60
Router	0.77	0.722	0.745	0.61
Portable Camera	0.69	0.72	0.70 8	0.70
Mobile Phone	0.69	0.76	0.72	0.66

TABLE II
EVALUATION OF FEATURE SCORING.

Review Domain	Accuracy
Video Player	0.69
Camera	0.83
Music Player	0.65
Portable Camera	0.77
Phone	0.79

is a feature based sentiment analysis system, and not a feature extraction system.

III. EVALUATION

The following formulae, from [11], will be employed for evaluation.

$$Precision = \frac{NumberofCorrect}{NumberofExtracted}$$

$$Recall = \frac{NumberofCorrect}{NumberofTrue}$$

$$F\text{-}measure = \frac{2 \times recall \times precision}{recall + precision}$$

$$Accuracy = \frac{CorrectOfQueries}{TotalQueries}$$

A. Feature Identification

The evaluation is done with the use of 'ground truths' for correct and incorrect because evaluation is done based on human perception and hence ranked as such, as opposed to clear mathematical precision of right or wrong.

5 corpus of reviews,taken from [5], belonging to different products were used to evaluate the system. The results of the system are shown below in table 1. The Comparable System refers to the results obtained by Subhabrata Mukherjee [3] using the same review corpus.

B. Sentiment Assignment to each Feature

The sentiment assignment forms the largest part of the proposed system. This section tests the proposed identification of opinion words and their corresponding polarity score.

Table 2 gives us the polarity classification correctness for each identified feature. The accuracy for each product would be higher if there is specific domain that the training set is from. So, a system well trained on mobile corpus positive / negative examples can be more effective in scoring a mobile domain corpus set.

IV. FUTURE WORK

A. Ontology

Using a domain knowledge system will improve feature identification. After a domain specific system is built, we can be sure that junk features will be discarded. On the other hand, features and their synonyms are also available to the analysis system to exploit. For example:

Worth the money

It is important to understand that the feature identification is linked in with the sentiment identification- that is in the absence of an associated opinion word- the sentiment system fails to identify the potential feature as a feature.

Worth the cost

Both cost and money are synonyms. But, in the absence of an ontology, both the words will be considered separate features. Much like the sentiment scores, the ontology must be generated prior to using the system for analysis, and stored for later use.

B. Status Array

Many researchers have remarked about the inability of existing systems to identify sarcasm [12]. This is a valid concern, and addressing this problem with an effective solution could help improve analysis by a great deal because angry reviewers often resort to sarcasm in their reviews.

Another issue is related to cross referencing nouns in sentence level analysis. For example: "The Speaker is great. It is loud." If the system knew that the noun in context was the 'speaker', it would have made an accurate classification that the speakers are loud.

To solve these problems, a review status array would be well suited. It could have multiple elements, but to simply deal with the two problems mentioned above:

```
<previous_noun;
previous_sentiment_polarity>
```

If no noun is identified in a sentence, the previously used (feature) noun would be used as the feature. Such a method implements continuity among the different sentences of a review.

Example for sarcasm handling:

The battery is great. It blew up on the second day.

Status Array for the above example,

```
<battery, positive>
```

Since, the second sentence is of negative polarity and references back to the previous noun, the system inverts the positive score assigned to the feature previously.

ISSN 2395-8618

C. Domain Pertinence

A potentially useful tool to filter out bad feature results would be using domain pertinence filter [13]. The same noise words are quite often found to populate multiple domains without belonging to one or the other. For example, the feature 'person' might be identified in both the agriculture and computer domain. In order to clean, we use the other domain's identified features as a filter.

V. Conclusion

Evaluation metrics would change dependent on the changes in tagging or parsing algorithm as well as the dataset used for training. The f-measure would be higher if the polarity training set was more relevant to the domain being analysed. The additions mentioned in the future work section could potentially give a significant improvement over the current system.

The outlined system could be used as a base upon which further improvements can be made. There is often a choice for the person building analysis system- he/she can opt for higher recall, and lower precision, or vice-versa. The designer will have to study the domain and requirements to achieve the targets for the project by striking a balance between recall and precision.

References

[1] X. Fang and J. Zhan, "Sentiment analysis using product review data," *Journal of Big Data*, vol. 2, no. 1, p. 1, 2015.

[11] D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, 1st ed. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2000.

[2] B. Agarwal, N. Mittal, P. Bansal, and S. Garg, "Sentiment analysis using common-sense and context information," *Computational intelligence and neuroscience*, vol. 2015, p. 30, 2015.

[3] S. Mukherjee and P. Bhattacharyya, "Feature specific sentiment analysis for product reviews," in *International Conference on Intelligent Text Processing and Computational Linguistics*. Springer, 2012, pp. 475–487.

[4] A. Collomb, C. Costea, D. Joyeux, O. Hasan, and L. Brunie, "A study and comparison of sentiment analysis methods for reputation evaluation."

[5] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2004, pp. 168–177.

[6] B. Ohana and B. Tierney, "Sentiment classification of reviews using SentiWordNet," in *9th. IT & T Conference*, 2009, p. 13.

[7] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093–1113, 2014.

[8] M. Radovanović and M. Ivanović, "Text mining: Approaches and applications," *Novi Sad J. Math*, vol. 38, no. 3, pp. 227–234, 2008.

[9] M.-C. de Marneffe and C. D. Manning, "The stanford typed dependencies representation," in *Coling 2008: Proceedings of the Workshop on Cross-Framework and Cross-Domain Parser Evaluation*, ser. CrossParser '08. Stroudsburg, PA, USA: Association for Computational Linguistics, 2008, pp. 1–8. [Online]. Available: http://dl.acm.org/citation.cfm?id=1608858.1608859

[10] M. Ganapathibhotla and B. Liu, "Mining opinions in comparative sentences," in *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*. Association for Computational Linguistics, 2008, pp. 241–248.

[12] K. Vivekanandan and J. S. Aravindan, "Aspect-based opinion mining: A survey," *International Journal of Computer Applications*, vol. 106, no. 3, 2014.

[13] J. De Knijff, F. Frasincar, and F. Hogenboom, "Domain taxonomy learning from text: The subsumption method versus hierarchical clustering," *Data & Knowledge Engineering*, vol. 83, pp. 54–69, 2013.