International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 06 | June - 2022 Impact Factor: 7.185 ISSN: 2582-3930

# SENTIMENT ANALYSIS USING PRODUCT REVIEW

C.Pavani, B.Mounika, D.Lavanya

**Abstarct:**

One of the main responsibilities of NLP is sentiment analysis or opinion mining. In recent days, sentiment analysis has received a lot of preference.

The goal of this study is to address the topic of sentiment polarity categorization, which is a difficult problem to solve the most basic problems in sentiment analysis a general method for With extensive process descriptions, sentiment polarity categorization is proposed Data.

product reviews from Amazon were used in this investigation. Experiment both sentence-level and review-level categorization are taken out.

with good results Finally, we provide some record into future work on

Analysis of how people feel.

**Keywords:**

Sentiment analysis; polarity separation of emotions; nlp reviews; processing.

## Introduction:

Sentiment is a feeling-driven attitude, idea, or judgement. Sentiment analysis , known as opinion mining, is the study of people's feelings about certain things.. From a user's view, they can submit their content on multiple social networking platforms.

Forums, microblogs, and online social networking sites are examples. Many social media networks make their APIs available to researchers ,

encouraging researchers and developers to acquire and analyse data For example, Twitter presently offers three API versions , notably the REST API, the SOAP API, and the Graph API.

Both the Search API and the Streaming API are available. The REST API allows developers to get status statistics and user information, while the Search API allows them to look up specific Twitter accounts.

On the other hand, the Streaming API can capture Twitter content in real time. Developers can also combine APIs to build their own apps. As a result, public opinion has shifted. Analysis looks to have a solid foundation thanks to the usage of large amounts of web data.

However, there are a number of drawbacks to using this type of internet data for sentiment research. People are free to contribute their own stuff, which is the first flaw.

The quality of their views, however, cannot be assured. Instead of, for example, Online spammers post spam on forums to share topic-related opinions. Some spam is unavoidable.

utterly meaningless, Others, on the other hand, have irrelevant or phoney opinions.

. The second issue is that such internet data does not mandatorily have a ground truth.

A ground truth is like a label for a particular view, indicating if or not the view is valid.

Positive, negative, or neutral are the three options.

| Star Level | General Meaning |
|---|---|
| ★ | I hate it. |
| ★★ | I don't like it. |
| ★★★ | It's okay. |
| ★★★★ | I like it. |
| ★★★★★ | I love it. |

datasets with ground truth that are also open to the public. There are 1.6 million machine-tagged Twitter messages in the corpus. Each communication is assigned a tag based on the emoticons (positive or negative) found inside it.

The data in this paper comes from a set of product reviews collected between February and April 2014 on Amazon . The issues listed above have been partially addressed in the following two ways: Before a product review can be published, it must first pass inspection. Second, each review must provide a rating that may be used as the basis for comparison. The ranking is based on a five-star system, with five stars being the highest and one star being the lowest (Figure 1).

This work focuses on sentiment polarity categorization , which is a key challenge in sentiment analysis. Figure 2 is a flowchart that shows the outline of this paper as well as our proposed categorization process. The majority of our contributions are in Phases 2 and 3. In Phase 2, 1) an algorithm for identifying negation phrases is proposed and implemented; 2) a mathematical approach for sentiment score computation is proposed; and 3) a feature vector generation method for sentiment polarity categorization is presented. Phase 3: 1) Two sentiment polarity categorization experiments are conducted at the phrase and review levels, respectively; 2) The performance of three classification models is evaluated and compared using the experimental findings.

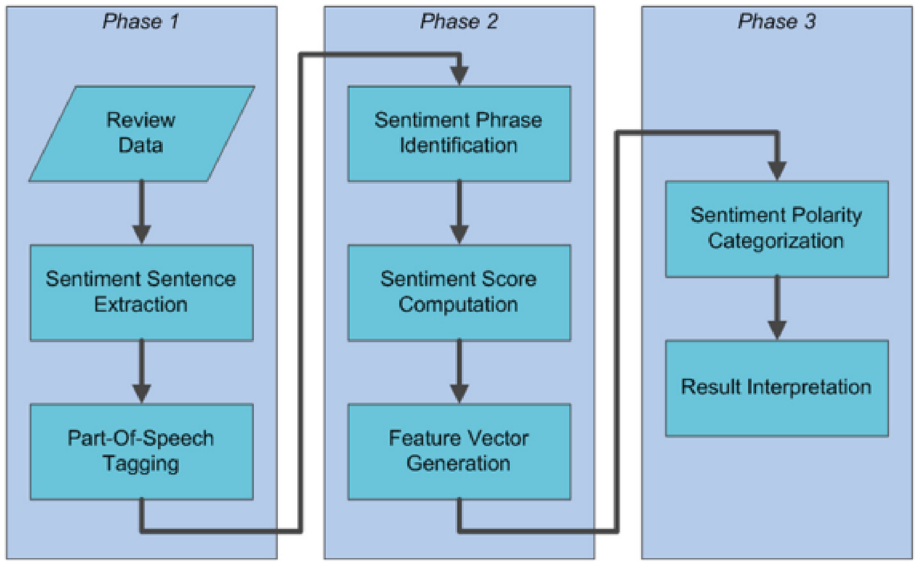



**Figure 2** Sentiment Polarity Categorization Process.

The following is how the rest of the paper is structured: We present a brief discussion of some relevant sentiment analysis work in the section 'Background and literature review.'

In the 'Methods' section, the software package and classification models utilised in this work are described. The section 'Background and literature review' proposes our thorough techniques to sentiment analysis. The findings of the experiments are reported in the section titled "Results and Discussion." The section 'Review-level categorization' contains discussion and future development. The 'Conclusion' section brings the paper to a close.

**Review of the background and literature**:

The categorization of sentiment polarity is a fundamental problem in sentiment analysis . The task is to categorise a piece of written language into one of two sentiment polarities: positive or negative (or neutral). There are three stages of emotion polarity categorization, depending on the breadth of the text: document level, sentence level, and entity and aspect level [26]. The document level is concerned with whether a document communicates negative or positive sentiment as a whole, whereas the sentence level is concerned with the sentiment categorization of each sentence; the entity and aspect level then focuses on what people like or dislike from their perspectives.

Since several reviews of sentiment analysis work have already been published , we will simply examine some past work in this area, on which our research is mostly based. Based on customer reviews, Hu and Liu summarised a list of positive terms and a list of negative words, respectively. There are 2006 words on the good list and 4783 terms on the negative list. Both lists also include some commonly misspelt words found in social media posts. Sentiment categorization is a classification problem in which features containing views or sentiment information must be found prior to classification. Pang and Lee [5] suggested removing objective statements and isolating subjective ones for feature selection. They presented a text-categorization technique that uses the minimum cut to identify subjective content. Gann et al. [28] chose 6,799 tokens from Twitter data, assigning each one a sentiment score, the TSI (Total Sentiment Index), indicating whether it is a positive or negative token. A token's TSI is calculated as follows: TSI = p -tp tn *n /p + tp/tn * n , where p is the number of times the token appears in positive tweets and n is the number of times the token appears in negative tweets. The ratio of total number of positive tweets to total number of negative tweets is called tp /tn.

Methodology and research design:

Obtaining information

This paper's data comes from a collection of Amazon.com product reviews. We collected almost 5.1 million product reviews between February and April 2014, with products falling into four primary categories: cosmetics, books, electronics, and home (Figure 3(a)). Over 3.2 million reviewers (customers) left over 3.2 million online reviews for a total of 20,062 products. The following information is included in each review: 1) reviewer ID; 2) product ID; 3) rating; 4) review duration; 5) usefulness; and 6) review text.

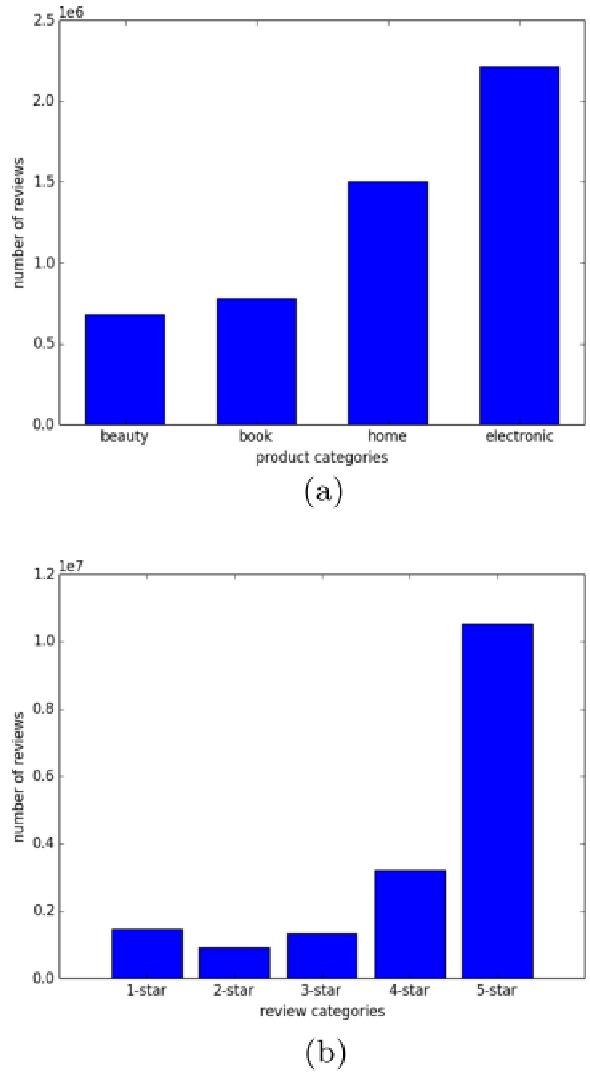



**Figure 3** Data collection **(a)** Data based on product categories **(b)** Data based on review categories.

Every rating is based on a 5-star scale (Figure 3(b)), so all ratings are from one to five stars, with no half-star or quarter-star in between.

Extraction of sentiment phrases and POS tagging

For sentiment analysis, Pang and Lee [5] recommend that all objective items be deleted. Rather of deleting objective content, all subjective content was extracted for future examination in our study. All sentiment sentences make up the subjective content. A sentiment sentence is one that has at least one word that is either good or negative. To begin, each sentence was tokenized into individual English words.

Every word in a phrase has a syntactic function that determines how it is utilised. The components of speech are another name for syntactic roles. The verb, the noun, the pronoun, the adjective, the adverb, the preposition, the conjunction, and the conjunction are the eight parts of speech in English.

**Table 1 Part-of-Speech tags for verbs**

| Tag | Definition |
|---|---|
| VB | base form |
| VBP | present tense, not 3rd person singular |
| VBZ | present tense, 3rd person singular |
| VBD | past tense |
| VBG | present participle |
| VBN | past participle |

Part-of-speech (POS) taggers [29-31] have been developed in natural language processing to classify words based on their parts of speech.

A POS tagger is particularly beneficial for sentiment analysis for the following two reasons:

1) Nouns and pronouns, for example, are frequently devoid of sentiment. A POS tagger can be used to filter out such terms; 2) A POS tagger can also be used to distinguish between words that can be used in various parts of speech. As an example, as a verb,

As an adjective, the word "enhanced" can convey a variety of emotions. The

The POS tagger employed in this study is a max-entropy POS tagger created for the Penn State University.

Project Treebank [31]. The tagger can provide 46 different tags, suggesting that it has a wide range of capabilities.Identify more detailed syntactic roles than the previous eight. Table 1 is an example of a list of all.

**tags for verbs that have been tagged with the POS tagger:**

The POS tagger was then used to tag each statement. Given the massive volume of data,In order to boost performance, a Python programme that can execute in parallel was built.The rate at which tags are added. As a result, there are more than 25 million adjectives and more than 22 million adverbs.Because there are over 56 million verbs labelled out of all the sentiment sentences, there are adverbs and adverbs.

Adjectives, adverbs, and verbs are all words that express emotion.

**Algorithm 1** Negation phrases identification

**Input:** Tagged Sentences, Negative Prefixes

**Output:** NOA Phrases, NOV Phrases

```
for every Tagged Sentences do
    for i/i+1 as every word/tag pair do
        if i+1 is a Negative Prefix then
            if there is an adjective tag or a verb tag in next pair then
                NOA Phrases ← (i, i+2)
                NOV Phrases ← (i, i+2)
            else
                if there is an adjective tag or a verb tag in the pair after next then
                    NOA Phrases ← (i, i+2, i+4)
                    NOV Phrases ← (i, i+2, i+4)
                end if
            end if
        end if
    end for
end for
return NOA Phrases, NOV Phrases
```

The POS tagger was then used to tag each statement. Given the large number of sentences, a Python programme that can run in parallel was built to speed up the tagging process. As a result, out of all the sentiment phrases, there are approximately 25 million adjectives, 22 million adverbs, and 56 million verbs labelled, because adjectives, adverbs, and verbs are the terms that primarily express sentiment.

Negative terms are used to identify people:

With the use of negative prefixes, words like adjectives and verbs can communicate the opposite sentiment. Consider the following line from a review of an electronic device: "The built-in speaker also has its uses, although nothing new so far." According to the list in [27], the word "revolutionary" is a positive word. However, the statement "nothing groundbreaking" conjures up mixed emotions. As a result, it is critical to recognise such sentences. Two types of phrases have been identified in this study: negation-of-adjective (NOA) and negation-of-verb (NOV) (NOV).

The POS tagger treats most common negative prefixes like not, no, and nothing as adverbs. As a result, we propose Algorithm 1 for phrase recognition.

The programme was able to find 21,586 different phrases, each with a negative prefix, with a total occurrence of over 0.68 million. Table 2 provides the top 5 NOA and NOV phrases in order of occurrence.

Calculation of sentiment scores for emotion tokens

A sentiment token is a single word or phrase that expresses emotion. A word token consists of a positive (negative) word and its part-of-speech tag, given the sentiment words described in [27]. We chose 11,478 word tokens in total, each of which appears at least 30 times in the dataset. From the 21,586 sentiment phrases discovered, 3,023 phrases were chosen for phrase tokens, with each of the 3,023 phrases having at least 30 occurrences. Given a token t, the formula for computing t's sentiment score (SS) is:

$$SS(t) = \frac{\sum_{i=1}^{5} i \times \gamma_{5,i} \times Occurrence_i(t)}{\sum_{i=1}^{5} \gamma_{5,i} \times Occurrence_i(t)}$$

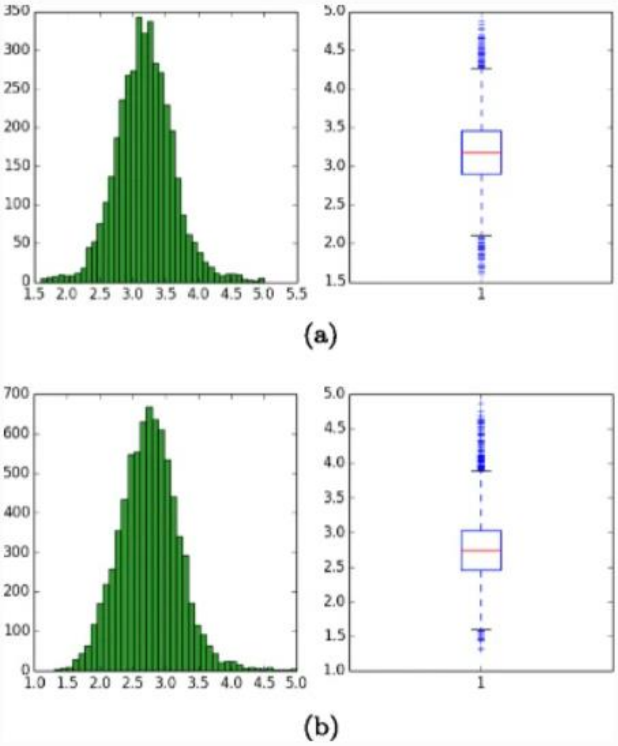
t's number of occurrence in i-star reviews is i(t), where i=1,...,5. Our dataset is not balanced, as seen in Figure 3, showing that different numbers of reviews were collected for each star level. Because 5-star ratings make up the majority of the data, we've created a ratio called 5,i, which is defined as:

$$\gamma_{5,i} = \frac{|5 - star|}{|i\text{-}star|}$$

The number of 5-star reviews is the numerator in equation 3, while the number of i-star reviews is the denominator, where i=1,...,5. As a result, if the dataset were balanced, 5,i would be set to 1 for each I in the dataset. As a result, every sentiment score should fall between the [1,5] range. We expect the median of positive word token sentiment ratings to be more than 3, which corresponds to the neutral point in Figure 1. It is reasonable to assume that the median for negative word tokens will be fewer than 3.

As a result, Figure 4 shows the emotion score information for positive word tokens (a). The histogram illustrates the distribution of scores, while the box plot demonstrates that the median is higher than the mean. Figure 4(b) demonstrates that the median sentiment score for negative word tokens is less than 3. In fact, the mean and median of positive word tokens both surpass 3, while both values for negative word tokens are less than 3. (Table 3).

**Figure 4**




Sentiment score information for word tokens **(a)** Positive word tokens **(b)** Negative word tokens.

## Table 3 Statistical information for word tokens

From: Sentiment analysis using product review data

| Token Type | Mean | Median |
|---|---|---|
| Positive Word Token | 3.18 | 3.16 |
| Negative Word Token | 2.75 | 2.71 |

**Labels that represent the truth on the ground**:

There are two stages to categorising sentiment polarity: sentence-level categorization and review-level categorization. The purpose of sentence-level classification is to categorise a sentence as good or negative based on the sentiment it communicates. Ground truth tags, identifying the positive or negative nature of a sentence, are required in the training data for this categorization procedure. However, due to the large amount of data we have, ground truth labelling becomes a very difficult challenge. Because manually tagging each sentence is impractical, a computer tagging approach is used instead. The method employs a bag-of-words model, which basically counts the number of positive or negative (word) tokens that exist in each phrase. If there are more positive tokens than negative tokens, the situation is favourable. This method is comparable to how the Sentiment 140 Tweet Corpus was tagged. The ground truth tags, which are the star-scaled ratings, are already present in the training data for review-level categorization.

Feature vector arrangement

Sentiment tokens and sentiment scores are derived from the original dataset's data. They're also known as features, and they'll be utilised to categorise sentiments. Each element of training data must be turned into a vector containing those features, referred to as a feature vector, in order to train the classifiers. A feature vector is created based on a sentence for sentence-level (review-level) categorization (review). Controlling the dimensionality of each vector is one problem. The task is actually divided into two parts: Because of the curse of dimensionality [32], a vector should not have an excessive number (thousands or hundreds) of features or values of a feature; second, each vector should have the same number of dimensions in order to match the classifiers. This problem is particularly acute in the case of sentiment tokens: On the one hand, there are 11,478 word tokens and 3,023 phrase tokens; on the other hand, vectors cannot be formed simply by including the tokens that appear in a sentence (or a review), because different sentences (or reviews) have different amounts of tokens, resulting in vectors of different dimensions.

Because we are only interested in each sentiment token's appearance within a sentence or a review, we utilise two binary strings to indicate each token's appearance. Word tokens are represented by an 11,478-bit string, while phrase tokens are represented by a 3,023-bit string. If the ith word (token) appears, the ith bit of the word (string) string will be switched from "0" to "1." Finally, rather than putting the flipped strings straight into a feature vector, each string's hash value is generated and saved using Python's built-in hash function. As a result, a sentence-level feature vector consists of four elements: two hash values computed using flipped binary strings, an averaged emotion score, and a sentence-level feature vector. In comparison, review-level vectors include one more element entirely. If there are m positive sentences and n negative sentences in a review, the element's value is computed as :-1m+1n.

## Results and discussion

### Evaluation methods

Performance of each classification model is estimated base on its averaged F1-score (4):

$$F1_{avg} = \frac{\sum_{i=1}^{n} \frac{2 \times P_i \times R_i}{P_i + R_i}}{n}$$

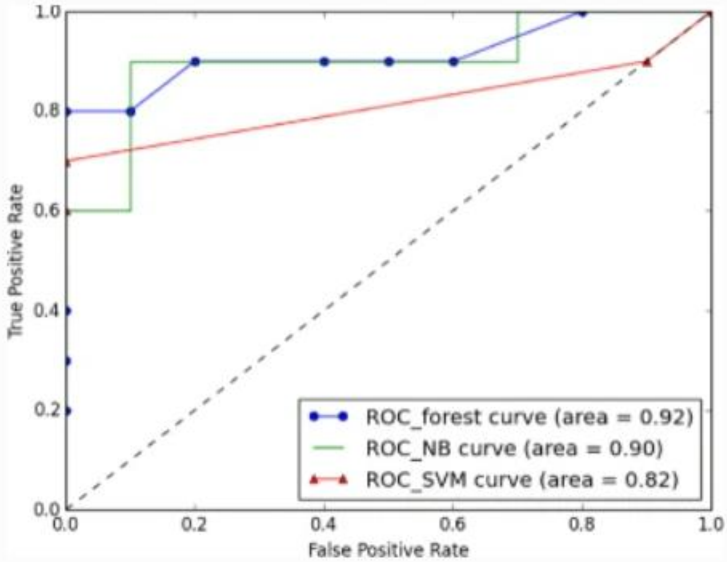
where P I is the ith class's precision, R I is the ith class's recall, and n is the number of classes 10-fold cross validation is used to evaluate P I and R i. The following is a 10-fold cross validation procedure: A dataset is divided into ten equal-sized subsets, each of which contains ten positive and ten negative class vectors. A single subset of the ten is kept as validation data for testing the classification model, while the other nine are used as training data. The cross-validation procedure is then done ten times, with each of the ten subsets serving as validation data exactly once. The ten folds' results are then added together. . Because the sentence-level categorization training data is categorised into two classes (positive and negative), ROC (Receiver Operating Characteristic) curves are plotted for a better performance comparison.

Categorization at the sentence level

The result is based on sentences that have been manually tagged.

Based on the 200 manually annotated sentences, 200 feature vectors are created. As a result, based on their F1-scores, the categorization models all perform at the same level, with the three scores all equaling 0.85. It is obvious from the ROC curves (Figure 5) that all three models performed admirably when evaluating data with a high posterior probability. (The classification model calculates a posterior probability of a testing data point, A, as the probability that A will be categorised as positive, indicated as P(+|A).) With a bigger area under curve, the Nave Bayesain classifier outperforms the SVM classifier as the probability decreases. The Random Forest is a type of random forest that is used to generate random numbers.

**Figure 5**

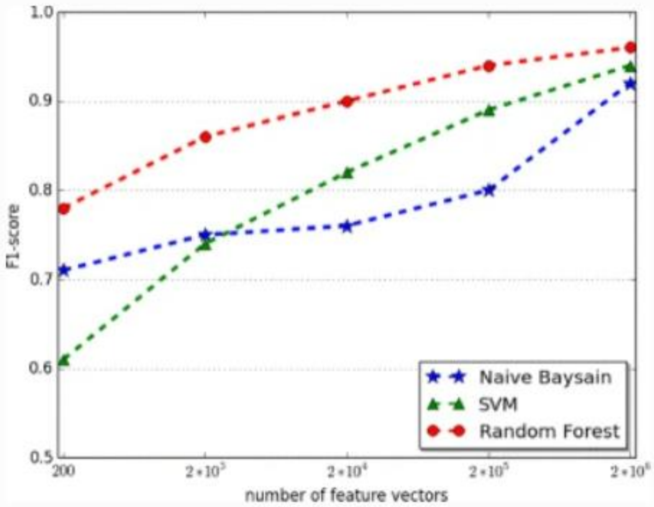



ROC curves based on the manually labeled set.

On the basis of machine-labeled sentences, arrive at a conclusion:

The whole collection generates 2-million feature vectors (1 million with positive labels and 1 million with negative labels) from 2-million machine-labeled texts. The whole collection is divided into four subsets: subset A contains 200 vectors, subset B contains 2,000 vectors, subset C contains 20,000 vectors, and subset D has 200,000 vectors. For each subset, the number of vectors with positive labels equals the number of vectors with negative labels. The categorization models' performance is then assessed using five distinct vector sets.

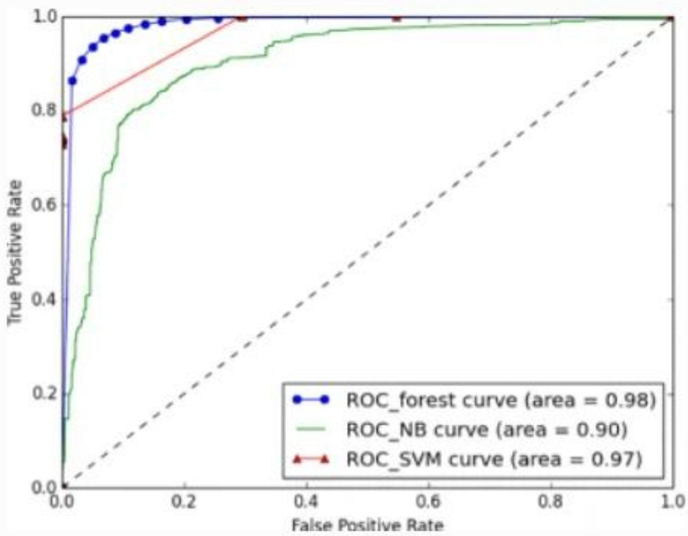
**Figure 6**




F1 scores of sentence-level categorization.

The F1 scores of the models are all increasing as more training data is added to them. As the training data expanded from 180 to 1.8 million, the SVM model had the most significant improvement, going from 0.61 to 0.94. On subset C and the entire set, the model surpasses the Nave Bayesain model and ranks as the second best classifier. The Random Forest model outperforms all other models for all datasets. Figure 7 depicts the ROC curves based on the whole set's results.

**Figure 7**



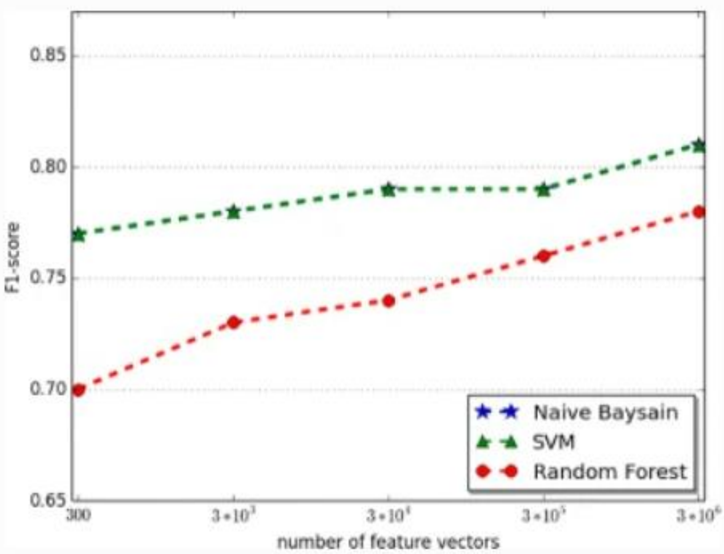

ROC curves based on the complete set.

**Categorization at the review level:**

For categorization, 3 million feature vectors are created. Positive vectors are formed from reviews with at least a 4-star rating, whilst negative vectors are generated from 1-star and 2-star reviews. To create neutral class vectors, 3-star reviews are employed. As a result, the entire set of vectors has been consistently labelled as positive, neutral, or negative. In addition, three subsets of the whole set are obtained, with subset A containing 300 vectors, subset B 3,000 vectors, subset C 30,000 vectors, and subset D 300,000 vectors, respectively.

The F1 scores obtained on various sizes of vector sets are shown in Figure 8. It is evident that the SVM model and the Nave Bayesain model are both correct. On all vector sets, both models perform better than the Random Forest model. However, because to their low performance on neutral class, neither model can achieve the same level of performance when employed for sentence-level classification.

**Figure 8**




F1 scores of review-level categorization.

Both in terms of sentence-level categorization and review-level categorization, the experimental outcome is positive. The averaged emotion score was shown to be a strong feature in all of itself, since it may reach an F1 score of over 0.8 for sentence-level categorization using the entire collection. The feature is capable of producing an F1 score of > 0.73 for review-level classification with the entire set. However, there are a few drawbacks to this research. The first is that if we wish to categorise reviews according to their individual star-scaled ratings, it becomes impossible to do so at the review level. To put it another way, the F1 scores achieved from such tests are typically low, with values less than 0.5. The second issue is that, because of our small size, we haven't been able to because the sentiment analysis scheme suggested in this paper is based on the presence of sentiment tokens, it may not be suitable for evaluations that only contain implicit sentiments. The polarity of an implicit sentiment is difficult to discern because it is usually transmitted through some neutral phrases. A statement like "Item as stated." for example, which regularly appears in positive reviews, is made up entirely of neutral words.

With those constraints in mind, we'll concentrate our efforts in the future on resolving those problems. To improve review-level categorizations, more features will be extracted and arranged into feature vectors. Our next step in the subject of implicit sentiment analysis is to be able to recognise the presence of such sentiment within the context of a certain product. Future work will include testing our categorization scheme using other datasets.

**Conclusion:**

Sentiment analysis, often known as opinion mining, is a branch of research that examines people's feelings, attitudes, and emotions about specific entities. Sentiment polarity categorization is a key challenge in sentiment analysis that is addressed in this study. Amazon.com product reviews were chosen as the source of data for this study. A process for categorising sentiment polarity (Figure 2) has been suggested, with thorough descriptions of each stage. Experiments on sentence-level categorization as well as review-level categorization were conducted.

**References**

1. Kim S-M, Hovy E (2004) Determining the sentiment of opinions In: Proceedings of the 20th international conference on Computational Linguistics, page 1367.. Association for Computational Linguistics, Stroudsburg, PA, USA.
2. Liu B (2010) Sentiment analysis and subjectivity In: Handbook of Natural Language Processing, Second Edition.. Taylor and Francis Group, Boca.
3. Liu B, Hu M, Cheng J (2005) Opinion observer: Analyzing and comparing opinions on the web In: Proceedings of the 14th International Conference on World Wide Web, WWW ’05, 342–351.. ACM, New York, NY, USA.
4. Pak A, Paroubek P (2010) Twitter as a corpus for sentiment analysis and opinion mining In: Proceedings of the Seventh conference on International Language Resources and Evaluation.. European Languages Resources Association, Valletta, Malta.
5. Pang B, Lee L (2004) A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts In: Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics, ACL ’04.. Association for Computational Linguistics, Stroudsburg, PA, USA.
6. Pang B, Lee L (2008) Opinion mining and sentiment analysis. Found Trends Inf Retr2(1-2): 1–135.
7. Turney PD (2002) Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, ACL ’02, 417–424.. Association for Computational Linguistics, Stroudsburg, PA, USA.
8. Whitelaw C, Garg N, Argamon S (2005) Using appraisal groups for sentiment analysis In: Proceedings of the 14th ACM International Conference on Information and Knowledge Management, CIKM ’05, 625–631.. ACM, New York, NY, USA.
9. Twitter (2014) Twitter apis. https://dev.twitter.com/start.
10. Liu B (2014) The science of detecting fake reviews. http://content26.com/blog/bing-liu-the-science-of-detecting-fake-reviews/.

11. Jindal N, Liu B (2008) Opinion spam and analysis In: Proceedings of the 2008 International Conference on, Web Search and Data Mining, WSDM ’08, 219–230.. ACM, New York, NY, USA.

12. Mukherjee A, Liu B, Glance N (2012) Spotting fake reviewer groups in consumer reviews In: Proceedings of the 21st, International Conference on World Wide Web, WWW ’12, 191–200.. ACM, New York, NY, USA.

13. Stanford (2014) Sentiment 140. http://www.sentiment140.com/.
14. www.amazon.com.
15. Go A, Bhayani R, Huang L (2009) Twitter sentiment classification using distant supervision, 1–12.. CS224N Project Report, Stanford.
16. Lin Y, Zhang J, Wang X, Zhou A (2012) An information theoretic approach to sentiment polarity classification In: Proceedings of the 2Nd Joint WICOW/AIRWeb Workshop on Web Quality, WebQuality ’12, 35–40.. ACM, New York, NY, USA.
17. Sarvabhotla K, Pingali P, Varma V (2011) Sentiment classification: a lexical similarity based approach for extracting subjectivity in documents. Inf Retrieval14(3): 337–353.
18. Wilson T, Wiebe J, Hoffmann P (2005) Recognizing contextual polarity in phrase-level sentiment analysis In: Proceedings of the conference on human language technology and empirical methods in natural language processing, 347–354.. Association for Computational Linguistics, Stroudsburg, PA, USA.
19. Yu H, Hatzivassiloglou V (2003) Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences In: Proceedings of the 2003 conference on, Empirical methods in natural language processing, 129–136.. Association for Computational Linguistics, Stroudsburg, PA, USA.
20. Zhang Y, Xiang X, Yin C, Shang L (2013) Parallel sentiment polarity classification method with substring feature reduction In: Trends and Applications in Knowledge Discovery and Data Mining, volume 7867 of Lecture Notes in Computer Science, 121–132.. Springer Berlin Heidelberg, Heidelberg, Germany.
21. Zhou S, Chen Q, Wang X (2013) Active deep learning method for semi-supervised sentiment classification. Neurocomputing120(0): 536–546. Image Feature Detection and Description.

**Article Google Scholar**

22. Chesley P, Vincent B, Xu L, Srihari RK (2006) Using verbs and adjectives to automatically classify blog sentiment. Training580(263): 233.

**Google Scholar**

23. Choi Y, Cardie C (2009) Adapting a polarity lexicon using integer linear programming for domain-specific sentiment classification In: Proceedings of the 2009 Conference on Empirical Methods in

Natural Language Processing: Volume 2 - Volume 2, EMNLP ’09, 590–598.. Association for Computational Linguistics, Stroudsburg, PA, USA.

Chapter Google Scholar

24. Jiang L, Yu M, Zhou M, Liu X, Zhao T (2011) Target-dependent twitter sentiment classification In: Proceedings of the 49th, Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, 151–160.. Association for Computational Linguistics, Stroudsburg, PA, USA.

Google Scholar

25. Tan LK-W, Na J-C, Theng Y-L, Chang K (2011) Sentence-level sentiment polarity classification using a linguistic approach In: Digital Libraries: For Cultural Heritage, Knowledge Dissemination, and Future Creation, 77–87.. Springer, Heidelberg, Germany.

Chapter Google Scholar

26. Liu B (2012) Sentiment Analysis and Opinion Mining. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.
27. Hu M, Liu B (2004) Mining and summarizing customer reviews In: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, 168–177.. ACM, New York, NY, USA.

Google Scholar

28. Gann W-JK, Day J, Zhou S (2014) Twitter analytics for insider trading fraud detection system In: Proceedings of the sencond ASE international conference on Big Data.. ASE.
29. Roth D, Zelenko D (1998) Part of speech tagging using a network of linear separators In: Coling-Acl, The 17th International Conference on Computational Linguistics, 1136–1142.
30. Kristina T (2003) Stanford log-linear part-of-speech tagger. http://nlp.stanford.edu/software/tagger.shtml.
31. Marcus M (1996) Upenn part of speech tagger. http://www.cis.upenn.edu/~treebank/home.html.
32. Han J, Kamber M, Pei J (2006) Data Mining: Concepts and Techniques, Second Edition (The Morgan Kaufmann Series in Data Management Systems), 2nd ed.. Morgan Kaufmann, San Francisco, CA, USA.
33. (2014) Scikit-learn. http://scikit-learn.org/stable/.