ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 8, Issue 8, August 2021

Sentiment Analysis Using Product Review Data

[1] Roshna Sanjana Kommareddy, [2] Dr. Y Mohana Roopa

[1][2] Department of Computer science and Engineering, Institute of Aeronautical Engineering, Hyderabad, India

Abstract---- **Analysis of the sentiments is one of the most important NLP jobs. It is frequently called as opinion mining. (Processing of natural language). In recent years, stimulus analysis has attracted considerable interest. The aim of this study is to solve one of the most basic challenges in sentiment analysis: categorising sentiment polarity. The authors offer a general method for categorising sentiment polarity, as well as detailed process descriptions. Data from online product reviews on Amazon.com was collected for this study. Experiments in the classification of sentences and the classification of reviews produced encouraging results. Finally, we provide some insight into our future attempts to analyse our feelings.**

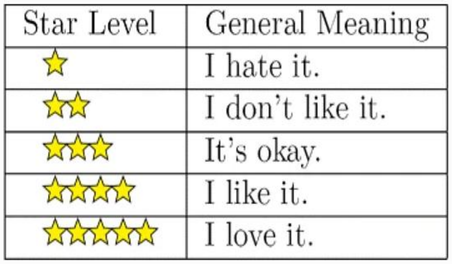
I. INTRODUCTION

A feeling-driven attitude, concept, or judgement is referred to as sentiment. The study of people's feelings on certain subjects is known as sentimental analysis (1-8). When it comes to feeling data the Internet is a wonderful resource. Individuals may submit their own material on different social media platforms such as social networking sites, microblogs and forums from the viewpoint of users. Many social networking networks provide academics and developers with their application programming (API) interfaces for collecting and analysing data. Three versions of the API [9], including the REST API, search and streaming API, are currently available on Twitter for example. The REST API is designed for developers to utilise to get information about status and user data. The Search API enables developers in general to search for Twitter while the Streaming API allows Twitter content to be captured. Developers may also mix APIs to create their own apps. As a result, feeling analysis seems to be based on large quantities of internet data.

However, there are a number of drawbacks to using this kind of internet data for sentiment research. The first flaw is that the quality of people's views cannot be ensured since they may post anything they wish. Installing online spammers in forums, for instance, instead of offering subject-related thoughts. Some spam are worthless while others include information that is incorrect or deceptive [10-12]. The second problem is that such information online does not always have a foundation. For a particular point of view, a basic truth is more like a label stating whether it's good, bad or neutral. One of the datasets that contains basic facts and is readily available to the public is the Stanford Sentiment 140 Tweet corpus. [13]. There are 1.6 million Twitter messages in the corpus that have been automatically labelled. Each communication is classified as good or bad based on the emoticons found within it.

The information included in this article is based on a compilation of product reviews from February to April 2014[14] on Amazon. To address the issues listed above, the following two approaches were utilised in part: An examination of a product should be carried out first prior to publication. Second, a rating to be used to compare goods must be given in each review. The 5-star system with the highest five stars and the lowest one star rating (Figure 1).

Figure 1

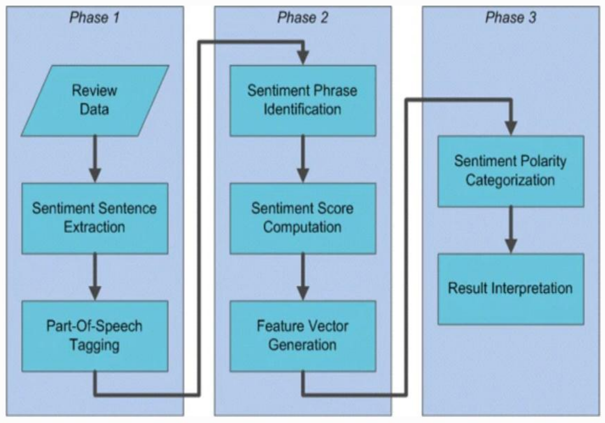


Star Level	General Meaning
★	I hate it.
★★	I don't like it.
★★★	It's okay.
★★★★	I like it.
★★★★★	I love it.

Rating System for Amazon.com.

The emphasis of this research is on sentiment polarity classification [15-21], which is a significant issue in sentiment analysis. Figure 2 is a flowchart that depicts the structure of this article as well as the classification method we suggest. Phases 2 and 3 are when the bulk of our contributions are made. 1) A technique for detecting negation phrases is given and implemented in the second phase. 2) The technique for creating vector is given for the categorization of emotional polarity. 3) A mathematical approach is recommended for computing sentiment ratings. In the third stage the performance of three classification models is measured and compared based on experimental results. Two feeling polarity classification testing are conducted at sentence and level of evaluation.

Figure 2


Phase 1
Phase 2
Phase 3
Review Data
Sentiment Sentence Extraction
Part-Of-Speech Tagging
Sentiment Phrase Identification
Sentiment Score Computation
Feature Vector Generation
Sentiment Polarity Categorization
Result Interpretation

Sentiment Polarity Categorization Process.

II. BACKGROUND AND LITERATURE REVIEW

Classification of feeling polarity is a key problem in the study of sentiment[6,22-25]. The objective is to categorise a text in one of two polarities of feeling: positive or negative (or neutral). The categorization of emotional polarity depends upon the length of the text: the level of the document, the phrase, the entity and the dimension level [26]. The document level addresses whether the whole text conveys a negative or a good feeling whereas the sentence level deals with each sentence's categorization. The entity level and aspect then addresses what people enjoy about their perspective or don't like about it.

Since assessments of many works have previously been discussed[26] under sentimental analysis, only some prior research in this area is being considered. The list of excellent keywords and the list of bad phrases based on customer assessments were summed up by the Hu and Liu[27]. There are 2006 terms in the positive list, whereas in the negative list 4783. These two lists also include many words which in social media postings are often misspelt. Features including views or feeling information, known as sentiment categorisation, must be discovered prior to classification. Pang and Lee[5] have suggested that target statements be eliminated and subjective isolated for feature selection. They developed a technique of text-categorization which defines the minimum content of the subjective. Gann et al. [28] chose 6.799 Twitter Tokens to indicate whether the token is positive and negative, each with a TSI (Total Sentiment Index). To calculate TSI of a token, the following formula is used:

$$TSI = \frac{p - \frac{tp}{tn} \times n}{p + \frac{tp}{tn} * n}$$

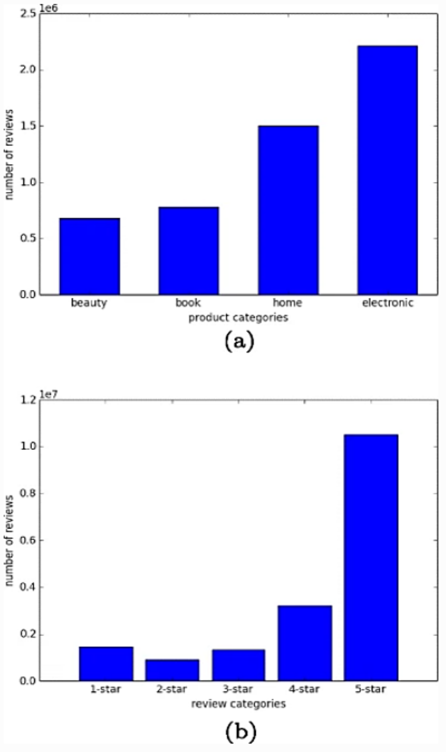
Where, p specifies how many times a token appears in excellent tweets and n how many times a token appears in poor tweets. tp/tn is the total positive to negative tweet ratio.

III. RESEARCH DESIGN AND METHDOLOGY

DATA COLLECTION

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), for a total of 20,062 items, have left over 3.2 million online review. Each review includes the following information: 1) the examiners' name; 2) the product name; 3) the rating, 4) the review length, 5) the usefulness, and 6) the language of review Both points are based on the 5-star (Figure 3(b)) scale and do not contradict half or fourth star.

Figure 3



(a)

(b)

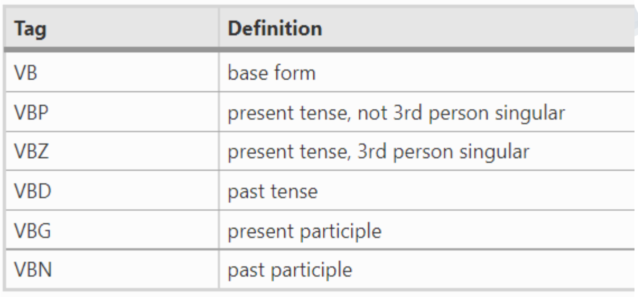
Data collection **(a)** Data based on product categories **(b)** Data based on review categories.

Sentiment SENTENCES extraction and POS tagging

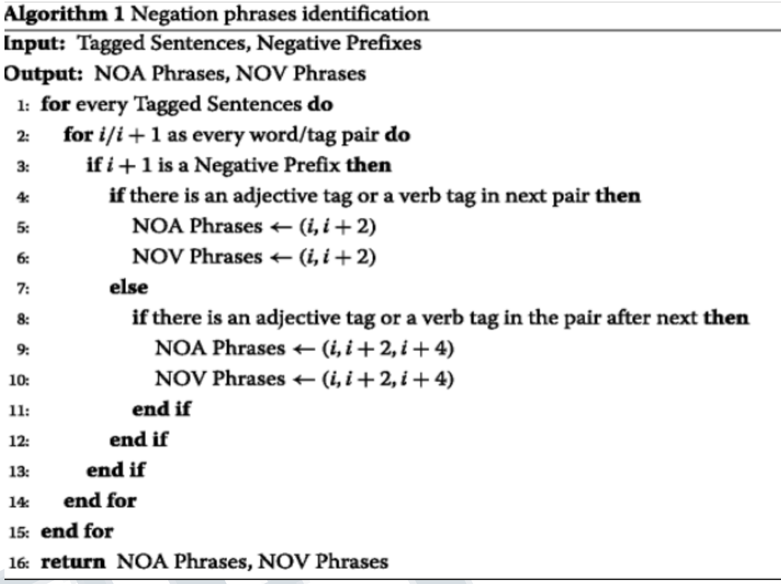
Before doing an analytical feeling, Pang and Lee [5] advise removing all objective factors. In order to better analyse our study, instead of deleting objective data, we removed all subjective material. There are all sentimental sentences in the subjective content. An emotional phrase includes a good or negative word at least. Each sentence was first tokenized into single English words.

Every word in the syntactic function of a sentence affects how it is used. Syntactic roles are also known as voice components. The eight language components are verb, substantive, pronoun, additional language, adverb, preposition, conjunction and interjection. In natural language processing, participatory language taggers [29-31] have been created to categorise words by components. A POS tagger is particularly helpful for feeling analysis for two reasons: 1) Signs and pronouns, for example, are frequently emotion-free. These keywords can be filtered using a POS tagging device. 2) A POS tagger may also be used to distinguish between the words to be used in distinct speech areas. 'Enhanced' as a verb may produce an emotional response that is different from an adjective 'enhanced.' In these research endeavours, the POS Tagger for Penn Treebank was utilised [31]. The tagger recognises 46 different syntactic roles, which indicates it can detect syntax roles more complicated than it can currently identify the eight roles. For example, Table 1 shows all verb tags in the POS tagger.

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

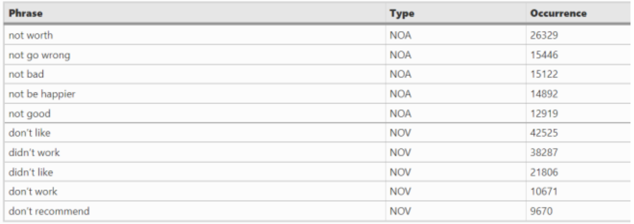


```
Algorithm 1 Negation phrases identification
Input: Tagged Sentences, Negative Prefixes
Output: NOA Phrases, NOV Phrases
for every Tagged Sentences do
    for $i/i+1$ as every word/tag pair do
        if $i+1$ is a Negative Prefix then
            if there is an adjective tag or a verb tag in next 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 used to tag each statement. Due to the large number of sentences, the tagging process was speededed up using the concurrent programme Pysthon. Because adjectives, adverbs and verbs are mainly the sensational phrase there, there are about 25 million Adjectives, 22 million Adverbs and 56 million verbs. Negative terms are used to identify people.

Words like adjectives and verbs may convey the opposite emotion by using negative prefixes. Consider this quote from an electronic gadget review: "There are other applications for the built-in speaker, but nothing new to date." According to the list in [27], the word "revolutionary" sprang to mind." is an encouraging word. The phrase "nothing groundbreaking" does nevertheless evoke contradictory emotions. It is thus essential that such sentences can be recognised. Two kind of sentences were discovered: denying the adjective (NOA) and negating the verb (NOV) This study revealed that (NOV).

The POS tagger is handled as adverbs most often with negative prefixes, such as not, no and nothing. As a result, Algorithm 1 is suggested for sentence recognition. Having a total incidence of over 0.68 million, the computer was able to identify 21,586 distinct sentences, each with a negative prefix. The first five sentences of NOA and NOV are presented in Table 2.



Phrase	Type	Occurrence
not worth	NOA	26329
not go wrong	NOA	15446
not bad	NOA	15122
not be happier	NOA	14892
not good	NOA	12919
don't like	NOV	42525
didn't work	NOV	38287
didn't like	NOV	21806
don't work	NOV	10671
don't recommend	NOV	9670

Sentiment score computation for sentiment tokens

A single word or phrase conveying emotion is called a symbol of feeling. In [27], the word token has a positive word and a speech tag, given the word emotions. We have chosen 11,478 word tokens which appear in the sample at least 30 times. 3,023 phrases were selected for phrase tokens from the 21,586 emotion phrases found, with each of the 3,023 phrases having at least 30 occurrences. A token t is the formula used to calculate the t sentiment score (SS)

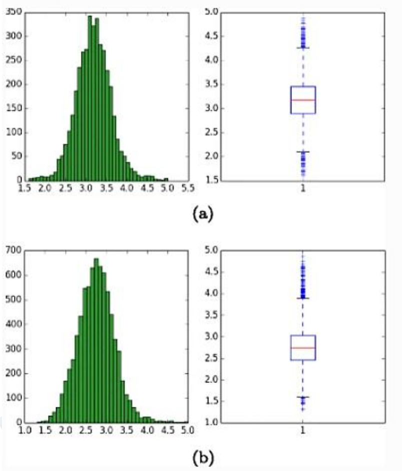
$$SS(t) = \frac{\sum_{i=1}^{5} i \times \gamma_{5,i} \times Occurrence_i(t)}{\sum_{i=1}^{5} \gamma_{5,i} \times Occurrence_i(t)}$$

The number of times t in i-star reviews occurs, where i=1,...,5 is i(t). Our dataset, as shown in Figure 3, is imbalanced, and various reviews are collected on every star level. Since 5-star ratings represent the overwhelming bulk of the information, we designed the 5-i ratio that is

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

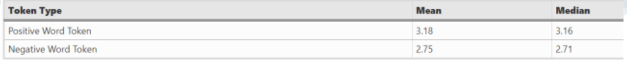
defined as:
The 5-star rating in Equation 3 is equivalent to the i-star rating, when i= 1,....,5 is the denominator. As a consequence, for each I in the dataset, 5,i would be set to 1 if the dataset was balanced. As a consequence, every emotion score should be in the range of [1,5]. The median positive word token emotion rating is expected to be more than 3, which corresponds to the neutral point in Figure 1. It's safe to anticipate that the median number of negative word tokens will be less than three.
Figure 4 shows the information on the emotional score for positive word tokens (a). This histogram indicates the scoring, while the median is above three in the case plot. The average value for bad word tokens is less than 3, as seen in Figure 4. (b). In reality, there are more than three median and mean positive terms, while fewer than three are median and mean negative words.

Figure 4



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

Table 3 Statistical information for word tokens



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

The ground truth labels

The polarity of the sentiment is divided into two stages: sentences and reviews. A sentence level is used to categorise a statement, based on the emotion it communicates, as good or negative. For this classification method, training data must include ground true tags which identify a specific phrase's positive or negative nature. Basic labelling of the truth on the other hand, because of the enormous amount of data that we have, becomes a difficult job. Since every sentence cannot be labelled manually, a computer-based tagging method has been employed. The technology depends upon a model bag-of-words, which counts for each phrase the number of positive and negative toks. When there are more positive tokens than negative ones, the statement is registered as positive. The technique of labelling the Tweet Corpus Sentiment 140 is similar. The basic truth tags are already available in the training data, which is star-scaled evaluations, for the classification of test levels.

Feature vector arrangement

The original dataset's data is used to generate sentiment tokens and sentiment scores. They're also called features, and they'll be used to classify emotions. Each training piece should be transformed into a feature vector containing those properties in order to train the classification devices. A vector is generated on the basis of a sentence at the sentence level (review level) (review). One issue is checking the dimensionality of each vector. Due to the curse dimension [32], no excessive quantities or values of the characteristic (thousands or hundreds) of characteristics or values should be included in a vector; secondly, to satisfy classifications, each vector must possess the same number of dimensions. This is extremely severe in emotional flags: On the contrary, tokens which exist in a sentence (or revision), since different sentences include diverse tokens, resulting in vectors of varying dimensions, may not just generate vectors. Only the tokens are created by adding tokens in a sentence (or revision).

We utilise two binary strings to denote the presence of each emotion token inside a phrase or a review since we are only interested in the appearance of each token within a sentence or a review. A 11,478-bit string represents a word token, whereas a 3,023-bit string represents a phrase token. The ith bit of the word (string) string will be altered from "0" to "1" if the ith word (token) occurs. Last but not least, the built-in Python hash function is utilised to generate and store a Hash value for each string rather than storing the flipped text directly in a feature vector. Thereby, a sentence-level feature vector consists of four parts: averaged emotion and a ground-reality label, two hazh values generated from flipped binary strings. On the other hand, review-level vectors have a complementary element. The value of the element is calculated in 1m+1n when the review includes m positive and n negative phrases.

IV. RESULTS AND DISCUSSION

Evaluation Methods

The averaged F1-score (4) of each classification model is used to measure its performance:

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

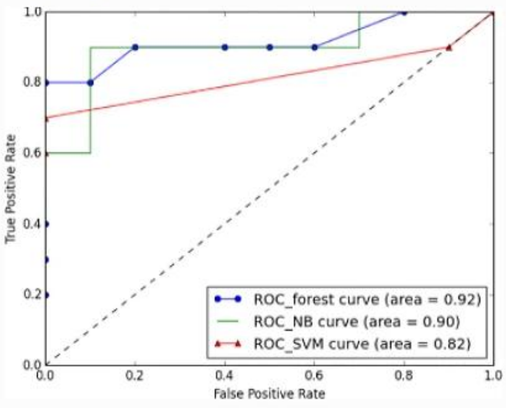
Where P I is the accuracy of class I, R I is class I and n is class I, and n is class numbers P I and R I validation 10 times across. The following is a 10 times cross-validation technique: Each of the 10 equal-sized subsets of a dataset contains ten positive and ten negative class vectors. The other nine are utilised for training, while validation data are retained as a single component of the 10 for the classification model. This process is performed 10 times and each of the ten sub-sets is utilised as validation data precisely once. The findings of the 10 folds were then averaged in order to provide one estimate. Curves for improved performance are presented as the phrase level classification data is split in two groups ROC(Receiver Operating Characteristic) (positive and negative).

Categorization at the sentence level

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

200 feature vectors are generated from the 200 carefully annotated phrases. As a result, the categorization models are all based on their F1 values at the same level, each of which equals 0.85 values. It is apparent that all three models were excellently executed by assessing data with a high postal probability from the ROC curves (Figure 5). (The chance that A will be classed positively as P(+|A) in the classification model is computed as the posterior probability of a test data point, A.) (A). When the likelihood declines, the Nave Bayesain classification surmounts the SVM classification since its range in curve is larger. In general, the Random Forest model is superior than the other.

Figure 5

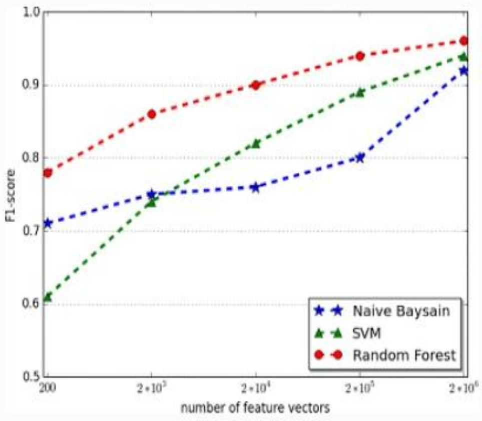


ROC curves based on the manually labeled set.

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

The collection generates 2 million vectors of machine labelling texts 2 million (1 million positive labels and one million negative ones). The subset of A consists of 200 vectors, the subset B consists of 2000, the subset of C includes 20 000, and the subset of C consists of 200 000 vectors. Subset C is made up of 200 000 vectors. The number of positive vectors for each subgroup equals the number of vectors with a negative labelling. The performance of the classification model is then evaluated using five different vector sets (four subsets and one complete set, Figure 6).

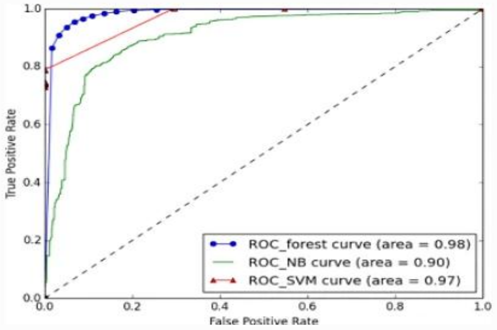
Figure 6



F1 scores of sentence-level categorization.

The F1 results are all improved when more training data is provided to the algorithms. With the training data up 180 million to 1,8 million, the SVM model improved greatly and went from 0.61 million to 0.94. The model ranks the second best classification on sub-set C and the whole collection. All other models for all data sets are overlapped by the Random Forest model. The ROC curves based on the whole data set are shown in Figure 7.

Figure 7



ROC curves based on the complete set.

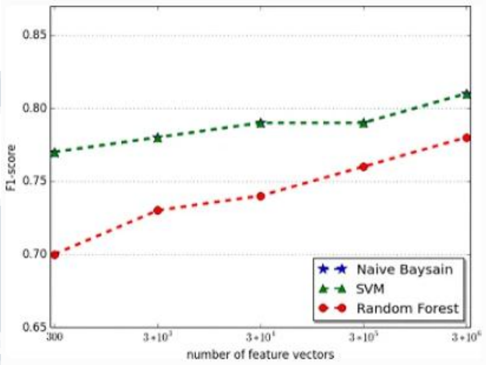
Review-Level Categorization

Classification generates 3 million feature vectors. Negative vectors are generated at 1-star and 2-star rates while positive vectors are generated at least 4-star rates from reviews. The generation of neutral vectors is based on 3-star ratings. As a result, the whole group of vectors is either positively, neutrally or negatively labelled.

Three sub-sets are produced of the entire package: the 300-vector subset A, the B-set 3000, the C-set 30,000 and the D-set 300,000.

The F1 results obtained on vector sets in various sizes are shown in figure 8. In terms of performance, the SVM and Nave Bayesain model are nearly comparable. On all vector sets of both models, the Random Forest model is better. Nevertheless, because of its poor neutral class performance neither model can attain the same performance when the sentence level is categorised.

Figure 8



F1 scores of review-level categorization.

In terms of both sentencing level classification and review level categorisation, the experimental findings are promising. Since the phrase level categorization using the whole collection may reach an F1 score over 0.8, its average emotional score has been a powerful feature. For review-level classification using the full set, the feature may provide an F1 score of > 0.73. However, there are a few flaws with this study. The first is that we can't do it at the review level, when we are going to categorise reviews based on their individual star ratings. In other words, the F1 values derived from these exams are often moderate and vary between 0.5 and 0.75. The second drawback is that, since our suggested technique of emotion analysis depends on the presence of feeling tokens, assessments with just implied feelings may not function effectively. Because implicit sentiments are typically conveyed via neutral words, it is difficult to determine their polarity. "Item as described." is a phrase that often occurs in good reviews and is completely composed of neutral phrases.

V. CONCLUSION

Analysis of sentiment, commonly referred to as Opinion Mining, is a kind of research that examines the feelings, views and emotions of individuals in regard to specific issues. Sentiment polarity classification is a major issue in sentiment analysis that this research tackles. The data for this research was gathered from Amazon.com product reviews. The technique for categorising feeling polarity with full clarifications of each stage is shown in Figure 2. Sentence-level classification and review-level categorization experiments were conducted.

Methods

In this research, a software programme for open source learning created in Python, Scikit-learn[33] was utilised. The categorization was based on the model classification of Bayesian ships, Random Forests, and Support Vector Machines[32].

Classification Bayesian naive

The classification of the Bayesian Nave works as follows: Suppose you have D-data, and the n-dimensional vectors represent every tuple, X=x 1,x 1,x 2,...,x n, that is, n measurements of n characteristics or attributes taken of the tuple. Suppose various classes are available (C 1, C 2,..., C m). Where I j,m, and j is both true, the classifier predicted that the tuple X would be true if: P(C i|X)> P(C j|X),

C I. The following formula is used to compute P(C i|X):

$$P(C_i|X) = \prod_{k=1}^{n} P(x_k|C_i)$$

Random Forest

The random forest classification was chosen because of the accuracy of a single decision tree. It's fundamentally a bagging-based ensemble method. The classifier works like: K bootstrap samples are produced from D by the classifier, each of which being marked D I D. The number of tuples sampled in the D I with D substitution is identical with D's. Some D tuples may not exist in D I because of the replacement sampling, while others may appear several times. The classifier then builds a decision tree on each D i. The k decision trees thus form a "forest." Each tree contributes one vote by guessing its class, which categorises an unknown tuple, X. In X's class, the person with the most votes gets to make the ultimate choice.

The decision tree method employed in scikit-learn is called CART (Classification and Regression Trees). The Gini index is used in CART's tree induction. This is how the Gini index for D is calculated:

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2$$

p I is likely to have Class C in a tuple in D I. The Gini index measures impurity. The lower the value of the index, the better divided D. Please read [32] for a full CART explanation.

Vector supporting machine

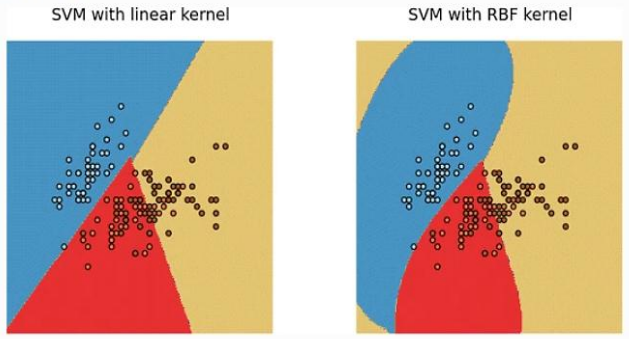
The support vector machine (SVM) for linear and non-linear data is a classification method. The SVM uses the linear optimal hyperplane separation (the linear kernel), a decision limit that divides data into groups when the data is divided linearly. For the splitting hyperplane, WX+b=0 is a mathematical equation, where W is a W=W, w2,...,w and X weighing vector is a training tuple. b is a variable of scalar. The issue essentially translates to the minimization of W in order to optimise the hyperplane, which is calculated as:i=1niyixii=1niyixi, where I are numeric parameters and y I are support vector-based labels, X i. That is: if y i=1 then ∑i=1nwixi≥1∑i=1nwixi≥1; if y i=−1 then ∑i=1nwixi≥−1∑i=1nwixi≥−1.

The SVM utilises nonlinear mapping to transfer data to a larger dimension if data is linearly inseparable. Then a linear hyperplane is found to deal with the problem. These changes are performed by the kernel functions. We utilised the Gaussian Radial Basis (RBF) as the kernel function for our experiment:

$$K(X_i, X_j) = e^{-\gamma\|X_i - X_j\|^2/2}$$

X j is testing tuples where X I support vectors, and in our trial there is a free parameter using a preset value for scikit-learn. The SVM grading examples using the linear kernel and kernel are shown in Figure 9.

Figure 9



A Classification Example of SVM.

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.