Expert Systems With Applications 57 (2016) 117–126

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

# Classification of sentiment reviews using n-gram machine learning approach

Abinash Tripathy*, Ankit Agrawal, Santanu Kumar Rath

*Department of Computer Science and Engineering, National Institute of Technology Rourkela, India*

ARTICLE INFO

*Article history:*
Received 3 March 2015
Revised 15 March 2016
Accepted 16 March 2016
Available online 24 March 2016

*Keywords:*
Sentiment analysis
Naive Bayes (NB)
Maximum Entropy (ME)
Stochastic Gradient Descent (SGD)
Support Vector Machine (SVM)
N-gram
IMDb dataset

ABSTRACT

With the ever increasing social networking and online marketing sites, the reviews and blogs obtained from those, act as an important source for further analysis and improved decision making. These reviews are mostly unstructured by nature and thus, need processing like classification or clustering to provide a meaningful information for future uses. These reviews and blogs may be classified into different polarity groups such as positive, negative, and neutral in order to extract information from the input dataset. Supervised machine learning methods help to classify these reviews. In this paper, four different machine learning algorithms such as Naive Bayes (NB), Maximum Entropy (ME), Stochastic Gradient Descent (SGD), and Support Vector Machine (SVM) have been considered for classification of human sentiments. The accuracy of different methods are critically examined in order to access their performance on the basis of parameters such as precision, recall, f-measure, and accuracy.



## 1. Introduction

Sentiment analysis, also known as opinion mining, analyzes people's opinion as well as emotions towards entities such as products, organizations, and their associated attributes. In the present day scenario, social media play a pertinent role in providing information about any product from different reviews, blogs, and comments. In order to derive meaningful information from people's sentiments, different machine learning techniques are applied by scholars and practitioners Liu (2012).

Sentiment analysis is observed to be carried out in three different levels such as document level, sentence level, and aspect level Feldman (2013). *Document level* classifies whether the document's opinion is positive, negative or neutral. *Sentence level* determines whether the sentence expresses any negative, positive or neutral opinion. *Aspect level* focuses on all expressions of sentiments present within given document and the aspect to which it refers. In this study, document level sentiment analysis has been taken into consideration.

There are mainly two types of machine learning techniques, which are very often used in sentiment analysis, i.e., the technique based on supervised and unsupervised learning. In supervised learning technique, the dataset is labeled and thus, trained to obtain a reasonable output which help in proper decision making Gautam and Yadav (2014). Unlike supervised learning, unsupervised learning process do not need any label data; hence they can not be processed at ease. In order to solve the problem of processing of unlabeled data, clustering algorithms are used Hastie, Tibshirani, and Friedman (2009). This study presents the impact of supervised learning method on labeled data.

The movie reviews are mostly in the text format and unstructured in nature. Thus, the stop words and other unwanted information are removed from the reviews for further analysis. These reviews goes through a process of vectorization in which, the text data are converted into matrix of numbers. These matrices are then given input to different machine learning techniques for classification of the reviews. Different parameters are then used to evaluate the performance of the machine learning algorithms.

The main contribution of the paper can be stated as follows:

i. Different machine learning algorithms are proposed for the classification of movie reviews of IMDb dataset IMDb (2011) using n-gram techniques viz., Unigram, Bigram, Trigram, combination of unigram and bigram, bigram and trigram, and unigram and bigram and trigram.
ii. Four different machine learning techniques such as Naive Bayes (NB), Maximum Entropy (ME), Support Vector Machine (SVM), and Stochastic Gradient Descent (SGD) are used for classification purpose using the n-gram approach.
iii. The performance of the machine leaning techniques are evaluated using parameters like precision, recall, f-measure, and

* Corresponding author. Tel.: +91 9437124235
*E-mail addresses:* abi.tripathy@gmail.com (A. Tripathy), agrawala96@gmail.com (A. Agrawal), skrath@nitrkl.ac.in (S.K. Rath).

http://dx.doi.org/10.1016/j.eswa.2016.03.028
0957-4174/

accuracy. The results obtained in this paper indicate, the higher values of accuracy when compared with studies made by other authors.

The structure of the paper is defined as follows: Section 2 presents literature survey. Section 3, indicates the methodology about the classification algorithm and its details. In Section 4, the proposed approach is explained. Section 5, indicates the implementation of the proposed approach. In Section 6, performance evaluation of the proposed approach is carried out. The last section i.e., Section 7 concludes the paper and presents the scope for future work.

## 2. Literature survey

The literature on sentiment analysis indicates that a good amount of study has been carried out by various authors based on document level sentiment classification.

### 2.1. Document level sentiment classification

Pang *et.al.*, have considered the aspect of sentiment classification based on categorization study, with positive and negative sentiments Pang, Lee, and Vaithyanathan (2002). They have undertaken the experiment with three different machine learning algorithms, such as, NB, SVM, and ME. The classification process is undertaken using the n-gram technique like unigram, bigram, and combination of both unigram and bigram. They have used bag-of-word features framework to implement the machine learning algorithms. As per their analysis, NB algorithm shows poor result among the three algorithms and SVM algorithm yields the result in a more convincing manner.

Salvetti *et.al.*, have discussed on Overall Opinion Polarity (OvOp) concept using machine learning algorithms such as NB and Markov model for classification Salvetti, Lewis, and Reichenbach (2004). In this paper, the hypernym provided by wordnet and Part Of Speech (POS) tag acts as lexical filter for classification. Their experiment shows that the result obtained by wordnet filter is less accurate in comparison with that of POS filter. In the field of OvOp, accuracy is given more importance in comparison with that of recall. In their paper, the authors presented a system where they rank reviews based on function of probability. According to them, their approach shows better result in case of web data.

Beineke *et.al.*, have used NB model for sentiment classification. They have extracted pair of derived features which are linearly combinable to predict the sentiment Beineke, Hastie, and Vaithyanathan (2004). In order to improve the accuracy result, they have added additional derived features to the model and used labeled data to estimate relative influence. They have followed the approach of Turney which effectively generates a new corpus of label document from the existing document Turney (2002). This idea allows the system to act as a probability model which is linear in logistics scale. The authors have chosen five positive and negative words as anchor words which produce 25 possible pairs and they used them for the coefficient estimation.

Mullen and Collier have applied SVM algorithm for sentiment analysis where values are assigned to few selected words and then combined to form a model for classification Mullen and Collier (2004). Along with this, different classes of features having closeness to the topic are assigned with the favorable values which help in classification. The authors have presented a comparison of their proposed approach with data, having topic annotation and hand annotation. The proposed approach has shown better result in comparison with that of topic annotation where as the results need further improvement, while comparing with hand annotated data.

Dave *et. al.* have used a tool for synthesizing reviews, then shifted them and finally sorted them using aggregation sites Dave, Lawrence, and Pennock (2003). These structured reviews are used for testing and training. From these reviews features are identified and finally scoring methods are used to determine whether the reviews are positive or negative. They have used a classifier to classify the sentences obtained from web-search through search query using product name as search condition.

Matsumoto *et.al.*, have used the syntactic relationship among words as a basis of document level sentiment analysis Matsumoto, Takamura, and Okumura (2005). In their paper, frequent word subsequence and dependency sub-trees are extracted from sentences, which act as features for SVM algorithm. They extract unigram, bigram, word subsequence and dependency subtree from each sentences in the dataset. They used two different datasets for conducting the classification i.e., IMDb dataset IMDb (2011) and Polarity dataset Pang and Lee (2004). In case of IMDb dataset, the training and testing data are provided separately but in Polarity dataset 10-fold cross validation technique is considered for classification as there is no separate data designated for testing or training.

Zhang *et.al.* have proposed the classification of Chinese comments based on word2vec and *SVM$^{perf}$* Zhang, Xu, Su, and Xu (2015). Their approach is based on two parts. In first part, they have used word2vec tool to cluster similar features in order to capture the semantic features in selected domain. Then in second part, the lexicon based and POS based feature selection approach is adopted to generate the training data. Word2vec tool adopts Continuous Bag-of-Words (CBOW) model and continuous skip-gram model to learn the vector representation of words Mikolov, Chen, Corrado, and Dean (2013). *SVM$^{perf}$* is an implementation of SVM for multi-variate performance measures, which follows an alternative structural formulation of SVM optimization problem for binary classification Joachims (2006).

Liu and Chen have proposed different multi-label classification on sentiment classification Liu and Chen (2015). They have used eleven multilevel classification methods compared on two micro-blog dataset and also eight different evaluation matrices for analysis. Apart from that, they have also used three different sentiment dictionary for multi-level classification. According to the authors, the multi-label classification process perform the task mainly in two phases i.e., problem transformation and algorithm adaptation Zhang and Zhou (2007). In problem transformation phase, the problem is transformed into multiple single-label problems. During training phase, the system learns from these transformed single label data, and in the testing phase, the learned classifier makes prediction at a single label and then translates it to multiple labels. In algorithm adaption, the data is transformed as per the requirement of the algorithm.

Luo *et.al.*, have proposed an approach to convert the text data into low dimension emotional space (ESM) Luo, Zeng, and Duan (2016). They have annotated small size words, which have definite and clear meaning. They have also used Ekman Paul's research to classify the words into six basic categories such as anger, fear, disgust, sadness, happiness and surprise Ekman and Friesen (1971). They again have considered two different approaches for assigning weight to words by emotional tags. The total weight of all emotional tags are calculated and based on these values, the messages are classified into different groups. Although their approach yields reasonably a good result for stock message board, the authors claim that it can be applied in any dataset or domain.

Niu *et.al.*, have proposed a Multi-View Sentiment Analysis (MVSA) dataset, including a set of image-text pair with manual annotation collected from Twitter Niu, Zhu, Pang, and El Saddik (2016). Their approach of sentiment analysis can be categorized into two parts, i.e., lexicon based and statistic learning. In case of lexicon based analysis, a set of opinion words or phrases are

**Table 1**
Comparison of sentiment techniques.

| Author | Approach | Algorithm used | Obtained result (Accuracy %) | Dataset used |
|---|---|---|---|---|
| Pang *et.al.* Pang et al. (2002) | Classify the dataset using different machine learning algorithms and n-gram model | Naive Bayes (NB), Maximum Entropy (ME), Support Vector Machine (SVM) | Unigram: SVM (82.9), Bigram: ME (77.4), Unigram + Bigram : SVM (82.7) | Internet Movie Database (IMDb) |
| Salvetti *et.al.* Salvetti et al. (2004) | Accessed overall opinion polarity(OvOp) concept using machine learning algorithms | Naive Bayes (NB) and Markov Model (MM) | NB: 79.5, MM: 80.5 | Internet Movie Database (IMDb) |
| Beineke *et.al.* Beineke et al. (2004) | Linearly combinable paired feature are used to predict the sentiment | Naive Bayes | NB: 65.9 | Internet Movie Database (IMDb) |
| Mullen and Collier Mullen and Collier (2004) | Values assigned to selected words then combined to form a model for classification | Support Vector Machine (SVM) | SVM: 86.0 | Internet Movie Database (IMDb) |
| Dave *et.al.* Dave et al. (2003) | Information retrieval techniques used for feature retrieval and result of various metrics are tested | $SVM^{lite}$, Machine learning using Rainbow, Naive Bayes | Naive Bayes : 87.0 | Dataset from Cnet and Amazon site |
| Matsumoto *et.al.* Matsumoto et al. (2005) | Syntactic relationship among words used as a basis of document level sentiment analysis | Support Vector Machine (SVM) | Unigram: 83.7, Bigram: 80.4, Unigram+Bigram : 84.6 | Internet Movie Database (IMDb), Polarity dataset |
| Zhang *et.al.* Zhang et al. (2015) | Use word2vec to capture similar features then classify reviews using $SVM^{perf}$ | $SVM^{perf}$ | Lexicon based: 89.95, POS based: 90.30 | Chinese comments on clothing products |
| Liu and Chen Liu and Chen (2015) | Used multi-label classification using eleven state-of-art multi-label, two micro-blog dataset, and eight different evaluation matrices on three different sentiment dictionaries. | Eight different evaluation matrices | Average highest Precision: 75.5 | Dalian University of Technology Sentiment Dictionary (DUTSD), National Taiwan University Sentiment Dictionary (NTUSD), Howset Dictionary (HD) |
| Luo *et.al.* Luo et al. (2016) | Ekman Paul's research approach is used to convert the text into low dimensional emotional space (ESM), then classify them using machine learning techniques Ekman and Friesen (1971) | Support Vector Machine (SVM), Naive Bayes (NB), Decision Tree (DT) | SVM: 78.31, NB: 63.28, DT: 79.21 | Stock message text data(The Lion forum) |
| Niu *et.al.* Niu et al. (2016) | Used Lexicon based analysis to transform data into required format and then use statistical learning methods to classify the reviews | BOW feature with TF and TF-IDF approach | Text: 71.9, Visual Feature: 68.7, Multi-view:75.2 | Manually annotated Twitter data |
| Proposed Approach | Converting text reviews into numeric matrices using countvectorizer and TF-IDF, which then given input to machine learning algorithms for classification | Support Vector Machine (SVM), Naive Bayes (NB), Maximum Entropy (ME), Stochastic Gradient Descent (SGD), N-gram | NB: 86.23, ME: 88.48, SVM: 88.94, SGD: 85.11 | Internet Movie Database (IMDb) |

considered which have pre-defined sentiment score. While in statistic learning, various machine learning techniques are used with dedicated textual features.

Table 1 provides a comparative study of different approaches adopted by different authors, contributed to sentiment classification.

### 2.2. Motivation for proposed approach

The above mentioned literature survey helps to identify some possible research areas which can be extended further. The following aspects have been considered for carrying out further research.

i. Most of the authors apart from Pang et al. (2002) and Matsumoto et al. (2005), have used unigram approach to classify the reviews. This approach provides comparatively better result, but fail in some cases. The comment "The item is not good," when analyzed using unigram approach, provides the polarity of sentence as neutral with the presence of one positive polarity word 'good' and one negative polarity word 'not'. But when the statement is analyzed using bigram approach, it gives the polarity of sentence as negative due to the presence of words 'not good', which is correct. Therefore, when a higher level of n-gram is considered, the result is expected to be better. Thus, analyzing the research outcome of several authors, this study makes an attempt to extend the sentiment classification using unigram, bigram, trigram, and their combinations for classification of movie reviews.

ii. Also a number of authors have used Part-of-Speech (POS) tags for classification purpose. But it is observed that the POS tag for a word is not fixed and it changes as per the context of their use. For example, the word 'book' can have the POS 'noun' when used as reading material where as in case of "ticket booking" the POS is verb. Thus, in order to avoid confusion, instead of using POS as a parameter for classification, the word as a whole may be considered for classification.

iii. Most of the machine learning algorithms work on the data represented as matrix of numbers. But the sentiment data are always in text format. Therefore, it needs to be converted to number matrix. Different authors have considered TF or TF-IDF to convert the text into matrix on numbers. But in this paper, in order to convert the text data into matrix of numbers, the combination of TF-IDF and CountVectorizer have been applied. The rows of the matrix of numbers represents a particular text file where as its column represent each word / feature present in that respective file which is shown in Table 3.

## 3. Methodology

Classification of sentiments may be categorized into two types, i.e., binary sentiment classification and multi-class sentiment classification Tang, Tan, and Cheng (2009). In binary classification type, each document $d_i$ in D, where D = $\{d_1, ......, d_n\}$ is classified as a label C, where C = {Positive, Negative} is a predefined category set. In multi class sentiment analysis, each document $d_i$ is classified as a label in $C^*$, where $C^*$ = {strong positive, positive, neural, negative, strong negative}. It is observed in the literature survey, that a good number of authors have applied binary classification method for sentiment analysis.

The movie reviews provided by the reviewers are mainly in text format; but for classification of sentiment of the reviews using the machine learning algorithms, numerical matrices are required. Thus, the task of conversion of text data in reviews into numerical matrices are carried out using different methods such as

- **CountVectorizer**: It converts the text document collection into a matrix of integers Garreta and Moncecchi (2013). This method helps to generate a sparse matrix of the counts.
- **Term frequency - Inverse document frequency (TF-IDF)**: It reflects the importance of a word in the corpus or the collection Garreta and Moncecchi (2013). TF-IDF value increases with increase in frequency of a particular word in the document. In order to control the generality of more common words, the term frequency is offset by the frequency of words in corpus. Term frequency is the number of times a particular term appears in the text. Inverse document frequency measures the occurrence of any word in all documents.

In this paper, the combination of methods i.e., CountVectorizer and TF-IDF have been applied to transform the text document into a numerical vector, which is then considered as input to supervised machine learning algorithm.

### 3.1. Application of machine learning algorithm

When supervised machine learning algorithms are considered for classification purpose, the input dataset is desired to be a labeled one. In this study, different supervised learning techniques are applied for classification purpose such as NB, ME, SGD, SVM, and n-gram method.

i. **Naive Bayes (NB) method:** This method is used for both classification as well as training purposes. This is a probabilistic classifier method based on Bayes' theorem. In this paper, multinomial Naive Bayes classification technique is used. Multinomial model considers word frequency information in document for analysis, where a document is considered to be an ordered sequence of words obtained from vocabulary 'V'. The probability of a word event is independent of word context and it's position in the document McCallum, Nigam et al. (1998). Thus, each document $d_i$ obtained from multinomial distribution of word is independent of the length of $d_i$. $N_{it}$ is the count of occurrence of $w_t$ in document $d_i$. The probability of a document belonging to a class, can be obtained using the following equation:

$$P(d_i|c_j;\theta) = P(|d_i|)|d_i|!\prod_{t=1}^{|V|}\frac{P(w_t|c_j;\theta)^{N_{it}}}{N_{it}!} \quad (1)$$

where $P(d_i|c_j;\ \theta)$ refers to the probability of document 'd' belonging to class 'c'. $P(|d_i|)$ is the probability of document 'd' and $P(w_t|c_j;\ \theta)$ is the probability of occurrence of a word 'w' in a class 'c'. After estimating the parameters calculated from training document, classification process is carried out on text document by calculating posterior probability of each class and selecting the highest probable class.

ii. **Maximum entropy (ME) method:** In this method, the training data is used to set constraint on conditional distribution Nigam, Lafferty, and McCallum (1999). Each constraint is used to express characteristics of training data. Maximum Entropy (ME) value in terms of exponential function can be expressed as

$$P_{ME}(c|d) = \frac{1}{Z(d)}exp(\sum_i \lambda_{i,c}f_{i,c}(d,c)) \quad (2)$$

where $P_{ME}(c|d)$ refers to probability of document 'd' belonging to class 'c', $f_{i,\ c}(d,\ c)$ is the feature / class function for feature $f_i$ and class c, $\lambda_{i,\ c}$ is the parameter to be estimated and Z(d) is the normalizing factor.
In order to use ME, a set of features is needed to be selected. For text classification purpose, word counts are considered as features. Feature / class function can be instantiated as follows:

$$f_{i,c'}(d,c) = \begin{cases} 0 & if c \neq c' \\ \frac{N(d,i)}{N(d)} & otherwise \end{cases} \quad (3)$$

where $f_{i,c'}(d,c)$ refers to features in word-class combination in class 'c' and document 'd', $N(d,\ i)$ represents the occurrence of feature 'i' in document 'd' and 'N(d)' number of words in 'd'. As per the expression, if a word occurs frequently in a class, the weight of word-class pair becomes higher in comparison to other pairs. These highest frequency word-class pairs are considered for classification purpose.

iii. **Stochastic gradient descent (SGD) method:** This method is used when the training data size is observed to be large. In SGD method instead of computing the gradient, each iteration estimates the value of gradient on the basis of single randomly picked example considered by Bottou (2012).

$$w_{t+1} = w_t - \gamma_t \nabla_W Q(z_t, w_t) \quad (4)$$

The stochastic process $\{w_t, t = 1, 2, .......\}$ depends on randomly picked example at each iteration, where $Q(z_t,\ w_t)$ is used to minimize the risk and $\gamma_t$ is the learning rate. The convergence of SGD gets effected by the noisy approximation of the gradient. If learning rate decreases slowly, the parameter estimate $w_t$ decreases equally slowly; but if rate decreases too quickly, the parameter estimate $w_t$ takes significant amount of time to reach the optimum point.

iv. **Support vector machine (SVM) method:** This method analyzes data and defines decision boundaries by having hyper-planes. In binary classification problem, the hyper-plane separates the document vector in one class from other class, where the separation between hyper-planes is desired to be kept as large as possible.
For a training set with labeled pair $(x_i, y_i), i = 1, 2, ....$ where $x_i \in R^n$ and y $\in \{1, -1\}^l$, the SVM method need to solve the following optimization problem, which can be represented as

$$\begin{aligned} &\min_{w,b,\xi} \ \tfrac{1}{2}W^TW + C\textstyle\sum_{i=1}^{l}\xi_i \\ &subject\ to\ y_i(w^T\phi(X_i)+b) \geq 1-\xi_i, \\ &\qquad \xi_i \geq 0. \end{aligned} \quad (5)$$

where 'W' is the weight parameter assigned to variables, $\xi$ is the slack or error correction added and 'C' is the regularization factor Hsu, Chang, and Lin (2003). Since the objective of the problem is to minimize "$\frac{1}{2}W^TW + C\sum_{i=1}^{l}\xi_i$," where value of "$y_i(w^T\phi(X_i)+b)$" needs to be greater than "1-$\xi_i$ " and the value of '$\xi$' is considered to be very small i.e., nearly equal to 0. Here training vector '$x_i$' is mapped to higher dimensional space by '$\phi$'.
Since SVM requires input in the form of a vector of numbers, the reviews of text file for classification need to be converted to numeric value. After the text file is converted to numeric vector, it may go through a scaling process, which helps to manage the vectors and keep them in the range of [1, 0].

v. **N-gram model:** It is a method of checking 'n' continuous words or sounds from a given sequence of text or speech. This model

**Table 2**
Confusion matrix.

| | Correct labels | |
|---|---|---|
| | Positive | Negative |
| Positive | TP (True positive) | FP (False positive) |
| Negative | FN (False negative) | TN (True negative) |

**Table 3**
Matrix generated under CountVectorizer scheme.

| | Feature 1 | Feature 2 | Feature 3 | Feature 4 | Feature 5 |
|---|---|---|---|---|---|
| Sentence 1 | 1 | 1 | 1 | 0 | 0 |
| Sentence 2 | 1 | 1 | 0 | 1 | 0 |
| Sentence 3 | 1 | 1 | 0 | 0 | 1 |

helps to predict the next item in a sequence. In sentiment analysis, the n-gram model helps to analyze the sentiment of the text or document. Unigram refers to n-gram of size 1, Bigram refers to n-gram of size 2, Trigram refers to n-gram of size 3. Higher n-gram refers to four-gram, five-gram, and so on. The n-gram method can be explained using following example:
A typical example of a sentence may be considered as "The movie is not a good one".

- Its unigram: "'The','movie','is', 'not', 'a', 'good','one'" where a single word is considered.
- Its bigram: "'The movie','movie is', 'is not', 'not a', 'a good', 'good one' " where a pair of words are considered.
- Its trigram: "'The movie is', 'movie is not', 'is not a', 'not a good','a good one'" where a set of words having count equal to three is considered.

### *3.2. Performance evaluation parameters*

The parameters helpful to evaluate performance of supervised machine learning algorithm is based on the element from a matrix known as confusion matrix or contingency table. It is used in supervised machine learning algorithm to help in assessing performance of any algorithm. From classification point of view, terms such as "True Positive(TP)", "False Positive (FP)", "True Negative(TN)", "False Negative (FP)" are used to compare label of classes in this matrix as shown in Table 2 Mouthami, Devi, and Bhaskaran (2013). True Positive represents the number of reviews those are positive and also classified as positive by the classifier, where as False Positive indicates positive reviews, but classifier does not classify it as positive. Similarly, True Negative represents the reviews which are negative also classified as negative by the classifier, where as False Negative are negative reviews but classifier does not classify it as negative.

Based on the values obtained from confusion matrix, other parameters such as "precision", "recall", "f-measure", and "accuracy" are found out for evaluating performance of any classifier.

- Precision: It measures the exactness of the classifier result. It is the ratio of number of examples correctly labeled as positive to total number of positively classified example.

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

- Recall: It measures the completeness of the classifier result. It is the ratio of total number of positively labeled example to total examples which are truly positive.

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

- F-Measure: It is the harmonic mean of precision and recall. It is required to optimize the system towards either precision or recall, which have more influence on final result.

$$F - Measure = \frac{2 \, * \, Precision \, * \, Recall}{Precision + Recall} \tag{8}$$

- Accuracy: It is the most common measure of classification process. It can be calculated as the ratio of correctly classified example to total number of examples.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

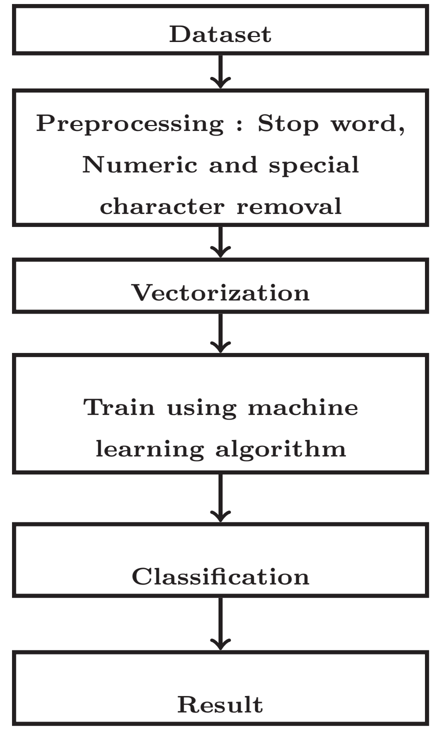


**Fig. 1.** Diagrammatic view of the proposed approach.

### *3.3. Dataset used*

In this paper, the acl Internet Movie Database (IMDb) dataset is considered for sentiment analysis IMDb (2011). It consists of 12,500 positively labeled test reviews, and 12,500 positively labeled train reviews. Similarly, there are 12,500 negative labeled test reviews, 12,500 negative labeled train reviews. Apart from labeled supervised data, an unsupervised dataset is also present with 50,000 unlabeled reviews.

## 4. Proposed-approach

The reviews of IMDb dataset is processed to remove the stop words and unwanted information from dataset. The textual data is then transformed to a matrix of number using vectorization techniques. Further, training of the dataset is carried out using machine learning algorithm. Steps of the approach is discussed in Fig. 1.

Step 1: The aclIMDb dataset consisting of 12,500 positive and 12,500 negative review for training and also 12,500 positive and 12,500 negative reviews for testing IMDb (2011), is taken into consideration.

Step 2: The text reviews sometimes consist of absurd data, which need to be removed, before considered for classification. The usually identified absurd data are:

- Stop words: They do not play any role in determining the sentiment.

- Numeric and special character: In the text reviews, it is often observed that there are different numeric (1,2,...5 etc.) and special characters (@, #, $,% etc.) present, which do not have any effect on the analysis. But they often create confusion during conversion of text file to numeric vector.

Step 3: After the preprocessing of text reviews, they need to be converted to a matrix of numeric vectors. The following methodologies are considered for conversion of text file to numeric vectors:

- CountVectorizer: It converts the text reviews into a matrix of token counts. It implements both tokenization and occurrence counting. The output matrix obtained after this process is a sparse matrix. This process of conversion may be explained using the following example:
  - Calculation of CountVectorizer Matrix: An example is considered to explain the steps of calculating elements of the matrix Garreta and Moncecchi (2013) which helps in improving the understandability. Suppose, three different documents containing following sentences are taken for analysis:
    Sentence 1: "Movie is nice".
    Sentence 2: "Movie is Awful".
    Sentence 3: "Movie is fine".
    A matrix may be formed with different values for its elements size 4*6, as there exists 3 documents and 5 distinct features. In the matrix given in Table 3, the elements are assigned with value of '1', if the feature is present or else in case of the absence of any feature, the element is assigned with value '0'.
- TF-IDF: It suggests the importance of the word to the document and whole corpus. Term frequency informs about the frequency of a word in a document and IDF informs about the frequency of the particular word in whole corpus Garreta and Moncecchi (2013).
  - Calculation of TF-IDF value: An example may be considered to improve understandability. If a movie review contains 1000 words wherein the word *"Awesome"* appears 10 times. The term frequency (i.e., TF) value for the word *"Awesome"* may be found as $10/1000 = 0.01$. Again, suppose there are 1 million reviews in the corpus and the word *"Awesome"* appears 1000 times in whole corpus. Then, the inverse document frequency (i.e., IDF) value is calculated as $\log(1{,}000{,}000/1{,}000) = 3$. Thus, the TF-IDF value is calculated as $0.01 * 3 = 0.03$.

Step 4: After the text reviews are converted to matrix of numbers, these matrices are considered as input for the following four different supervised machine learning algorithms for classification purpose.

- NB method: Using probabilistic classifier and pattern learning, the set of documents are classified McCallum et al. (1998).
- ME method: The training data are used to set constraint on conditional distribution Nigam et al. (1999). Each constraint is used to express characteristics of training the data. These constraints then are used for testing the data.
- SGD method: SGD method is used when the training data size is mostly large in nature. Each iteration estimates the gradient on the basis of single randomly picked example Bottou (2012).
- SVM method: Data are analyzed and decision boundaries are defined by having hyper planes. In two category case, the hyper plane separates the document vector of one class from other classes, where the separation is maintained to be large as possible Hsu et al. (2003).

Step 5: As mentioned in step 1, the movie reviews of acl IMDb dataset is considered for analysis, using the machine learning algorithms discussed in step 4. Then different variation of the n-gram methods i.e., unigram, bigram, trigram, unigram + bigram, unigram + trigram, and unigram + bigram + trigram are applied to obtain the result which is shown in Section 5.

Step 6: The results obtained from this analysis are compared with the results available in other literatures is shown in Section 6.

## 5. Implementation

- **Application of NB method:** The confusion matrix and various evaluation parameters such as precision, recall, f-measure, and accuracy values obtained after classification using NB n-gram techniques are shown in Table 4.
  As shown in Table 4, it can be analyzed that the accuracy value obtained using bigram is better than value obtained using techniques such as unigram and trigram. NB method is a probabilistic method, where the features are independent of each other. Hence, when analysis is carried out using "single word (unigram)" and "double word (bigram)", the accuracy value obtained is comparatively better than that obtained using trigram. But when 'triple word (trigram)' is being considered for analysis of features, words are repeated a number of times; thus, it affects the probability of the document. For example: for the statement "it is not a bad movie", the trigram "it is not", and "is not a" show negative polarity, where as the sentence represents positive sentiment. Thus, the accuracy of classification decreases. Again, when the trigram model is combined with unigram or bigram or unigram + bigram, the impact of trigram makes the accuracy value comparatively low.
- **Application of ME method:** The confusion matrix and evaluation parameters such as precision, recall, f-measure, and accuracy values obtained after classification using ME n-gram techniques are shown in Table 5.
  As represented in the Table 5, it may be analyzed that the accuracy value obtained using unigram is better than that of bigram and trigram. As ME algorithm based on conditional distribution and word-class pair help to classify the review, unigram method which considers single word for analysis, provides best result in comparison with other methods. In both bigram and trigram methods, the negative or positive polarity word appears more than once; thus, affecting the classification result. The bigram and trigram methods when combined with unigram and between themselves, the accuracy values of various combinations are observed to be low.
- **Application of SVM method:** The confusion matrix and evaluation parameters such as precision, recall, f-measure, and accuracy values obtained after classification using SVM n-gram techniques are shown in Table 6.
  As exhibit in Table 6, it may be analyzed that the accuracy value obtained using unigram is better than the value obtained using bigram and trigram. As SVM method is a non-probabilistic linear classifier and trains model to find hyperplane in order to separate the dataset, the unigram model which analyzes single words for analysis gives better result. In bigram and trigram, there exists multiple word combinations, which, when plotted in a particular hyperplane, confuses the classifier and thus, it provides a less accurate result in comparison with the value obtained using unigram. Thus, the less ac-

**Table 4**
Confusion matrix, evaluation parameter and accuracy for naive bayes n-gram classifier.

| Method | | Confusion matrix | | Evaluation parameter | | | Accuracy |
|---|---|---|---|---|---|---|---|
| Unigram | | Correct labels | | Precision | Recall | F-Measure | 83.652 |
| | | Positive | Negative | | | | |
| | Positive | 11025 | 1475 | 0.88 | 0.81 | 0.84 | |
| | Negative | 2612 | 9888 | 0.79 | 0.87 | 0.83 | |
| Bigram | | Correct labels | | Precision | Recall | F-Measure | 84.064 |
| | | Positive | Negative | | | | |
| | Positive | 11156 | 1344 | 0.89 | 0.81 | 0.85 | |
| | Negative | 2640 | 9860 | 0.79 | 0.88 | 0.83 | |
| Trigram | | Correct labels | | Precision | Recall | F-Measure | 70.532 |
| | | Positive | Negative | | | | |
| | Positive | 10156 | 2344 | 0.81 | 0.67 | 0.73 | |
| | Negative | 5023 | 7477 | 0.6 | 0.76 | 0.67 | |
| Unigram + Bigram | | Correct labels | | Precision | Recall | F-Measure | 86.004 |
| | | Positive | Negative | | | | |
| | Positive | 11114 | 1386 | 0.89 | 0.84 | 0.85 | |
| | Negative | 2113 | 10387 | 0.83 | 0.88 | 0.85 | |
| Bigram + Trigram | | Correct labels | | Precision | Recall | F-Measure | 83.828 |
| | | Positive | Negative | | | | |
| | Positive | 11123 | 1377 | 0.89 | 0.81 | 0.85 | |
| | Negative | 2666 | 9834 | 0.79 | 0.88 | 0.83 | |
| Unigram + Bigram + Trigram | | Correct labels | | Precision | Recall | F-Measure | 86.232 |
| | | Positive | Negative | | | | |
| | Positive | 11088 | 1412 | 0.89 | 0.85 | 0.87 | |
| | Negative | 2030 | 10470 | 0.84 | 0.88 | 0.86 | |

**Table 5**
Confusion matrix, evaluation parameter and accuracy for maximum entropy n-gram classifier.

| Method | | Confusion matrix | | Evaluation parameter | | | Accuracy |
|---|---|---|---|---|---|---|---|
| Unigram | | Correct labels | | Precision | Recall | F-Measure | 88.48 |
| | | Positive | Negative | | | | |
| | Positive | 11011 | 1489 | 0.88 | 0.89 | 0.88 | |
| | Negative | 1391 | 11109 | 0.89 | 0.88 | 0.88 | |
| Bigram | | Correct labels | | Precision | Recall | F-Measure | 83.228 |
| | | Positive | Negative | | | | |
| | Positive | 10330 | 2170 | 0.83 | 0.84 | 0.83 | |
| | Negative | 2023 | 10477 | 0.84 | 0.83 | 0.83 | |
| Trigram | | Correct labels | | Precision | Recall | F-Measure | 71.38 |
| | | Positive | Negative | | | | |
| | Positive | 8404 | 4096 | 0.67 | 0.73 | 0.70 | |
| | Negative | 3059 | 9441 | 0.76 | 0.70 | 0.73 | |
| Unigram + Bigram | | Correct labels | | Precision | Recall | F-Measure | 88.42 |
| | | Positive | Negative | | | | |
| | Positive | 11018 | 1482 | 0.88 | 0.89 | 0.88 | |
| | Negative | 1413 | 11087 | 0.89 | 0.88 | 0.88 | |
| Bigram + Trigram | | Correct labels | | Precision | Recall | F-Measure | 82.948 |
| | | Positive | Negative | | | | |
| | Positive | 10304 | 2196 | 0.82 | 0.83 | 0.83 | |
| | Negative | 2067 | 10433 | 0.83 | 0.83 | 0.83 | |
| Unigram + Bigram + Trigram | | Correct labels | | Precision | Recall | F-Measure | 83.36 |
| | | Positive | Negative | | | | |
| | Positive | 11006 | 1494 | 0.88 | 0.89 | 0.88 | |
| | Negative | 2666 | 9834 | 0.78 | 0.87 | 0.82 | |

curate bigram and trigram, when combined with unigram and with each other also, provide a less accurate result.

- **Application of SGD method:** The confusion matrix and evaluation parameters such as precision, recall, f-measure, and accuracy values obtained after classification using SGD n-gram techniques are shown in Table 7.

As illustrate in Table 7, it can be analyzed that the accuracy obtained using unigram is better than that of bigram and trigram. In SDG method, the gradient is estimated on single randomly picked reviews using learning rate to minimize the risk. In unigram, a single word is randomly picked to analyze, but in bigram and trigram both the combination of the words adds noise, which reduces the value of accuracy. Thus, when the bigram and trigram model is combined with other model, their less accuracy value affects the accuracy of the total system.

## 6. Performance evaluation

The comparative analysis based on results obtained using proposed approach to that of other literatures using IMDb dataset and n-gram approaches are shown in Table 8.

Pang et. al., have used machine learning algorithm viz., NB, ME method, and SVM method using n-gram approach of unigram, bigram and combination of unigram and bigram. Salvetti *et.al.* and Beineke *et.al.* have implemented the NB method for classification; but only the unigram approach is used for classification. Mullen and Collier, have proposed SVM method for classification; with unigram approach only. Matsumoto *et.al.* have also implemented the SVM for classification and used the unigram, bigram, and combination of both i.e., unigram and bigram for classification.

**Table 6**
Confusion matrix, evaluation parameter and accuracy for support vector machine n-gram classifier.

| Method | Confusion matrix | | | Evaluation parameter | | | Accuracy |
|---|---|---|---|---|---|---|---|
| unigram | | Correct labels | | Precision | Recall | F-Measure | 86.976 |
| | | Positive | Negative | | | | |
| | Positive | 10993 | 1507 | 0.88 | 0.86 | 0.87 | |
| | Negative | 1749 | 10751 | 0.86 | 0.88 | 0.87 | |
| Bigram | | Correct labels | | Precision | Recall | F-Measure | 83.872 |
| | | Positive | Negative | | | | |
| | Positive | 10584 | 1916 | 0.85 | 0.83 | 0.84 | |
| | Negative | 2116 | 10384 | 0.83 | 0.84 | 0.84 | |
| Trigram | | Correct labels | | Precision | Recall | F-Measure | 70.204 |
| | | Positive | Negative | | | | |
| | Positive | 8410 | 4090 | 0.67 | 0.71 | 0.69 | |
| | Negative | 3359 | 9141 | 0.73 | 0.69 | 0.71 | |
| Unigram +Bigram | | Correct labels | | Precision | Recall | F-Measure | 88.884 |
| | | Positive | Negative | | | | |
| | Positive | 11161 | 1339 | 0.89 | 0.89 | 0.89 | |
| | Negative | 1440 | 11060 | 0.88 | 0.89 | 0.89 | |
| Bigram + Trigram | | Correct labels | | Precision | Recall | F-Measure | 83.636 |
| | | Positive | Negative | | | | |
| | Positive | 10548 | 1152 | 0.84 | 0.83 | 0.84 | |
| | Negative | 2139 | 10361 | 0.83 | 0.84 | 0.84 | |
| Unigram + Bigram + Trigram | | Correct labels | | Precision | Recall | F-Measure | 88.944 |
| | | Positive | Negative | | | | |
| | Positive | 11159 | 1341 | 0.89 | 0.89 | 0.89 | |
| | Negative | 1423 | 11077 | 0.89 | 0.89 | 0.89 | |

**Table 7**
Confusion matrix, Evaluation parameter and accuracy for stochastic gradient descent n-gram classifier.

| Method | Confusion matrix | | | Evaluation parameter | | | Accuracy |
|---|---|---|---|---|---|---|---|
| Unigram | | Correct labels | | Precision | Recall | F-Measure | 85.116 |
| | | Positive | Negative | | | | |
| | Positive | 9860 | 2640 | 0.79 | 0.90 | 0.84 | |
| | Negative | 1081 | 11419 | 0.91 | 0.81 | 0.86 | |
| Bigram | | Correct labels | | Precision | Recall | F-Measure | 95 |
| | | Positive | Negative | | | | |
| | Positive | 12331 | 169 | 0.99 | 0.92 | 0.95 | |
| | Negative | 1081 | 11419 | 0.91 | 0.99 | 0.95 | |
| Trigram | | Correct labels | | Precision | Recall | F-Measure | 58.408 |
| | | Positive | Negative | | | | |
| | Positive | 11987 | 513 | 0.96 | 0.55 | 0.70 | |
| | Negative | 9885 | 2615 | 0.21 | 0.84 | 0.33 | |
| Unigram + Bigram | | Correct labels | | Precision | Recall | F-Measure | 83.36 |
| | | Positive | Negative | | | | |
| | Positive | 9409 | 3091 | 0.75 | 0.90 | 0.82 | |
| | Negative | 1069 | 11431 | 0.91 | 0.79 | 0.85 | |
| Bigram + Trigram | | Correct labels | | Precision | Recall | F-Measure | 58.744 |
| | | Positive | Negative | | | | |
| | Positive | 12427 | 73 | 0.99 | 0.55 | 0.71 | |
| | Negative | 10241 | 2259 | 0.18 | 0.97 | 0.30 | |
| Unigram + Bigram + Trigram | | Correct labels | | Precision | Recall | F-Measure | 83.336 |
| | | Positive | Negative | | | | |
| | Positive | 9423 | 3077 | 0.75 | 0.90 | 0.82 | |
| | Negative | 1089 | 11411 | 0.91 | 0.79 | 0.85 | |

In this present paper, four different algorithms viz., NB, ME method, SVM, and SGD using n-gram approaches like unigram, bigram, trigram, unigram+bigram, bigram+trigram, and unigram+bigram+trigram are carried out. Result obtained in the present approach is observed to be better than the result available in the literature where both IMDb dataset and n-gram approach are used.

### *6.1. Managerial insights based on result*

The managerial insight based on the obtained result can be explained as follows:

- It was almost an observed practice that, sellers send questionnaires to the customers, about the feed back of the product they have bought. But now-a-days people share those views through reviews or blogs.
- The reviews can be collected and given input to the proposed approach for qualitative decisions.
- The proposed approach classifies the reviews into either positive or negative polarity; hence is able to guide the managers properly by informing them about the shortcoming or good features of the product which they need to incorporate, to sustain the market competition.

## 7. Conclusion and future work

This paper makes an attempt to classify movie reviews using various supervised machine learning algorithms, such as Naive Bayes (NB), Maximum Entropy (ME), Stochastic Gradient De-

**Table 8**
Comparative result of values on “Accuracy” result obtained with different literature using IMDb Dataset and ngram approach.

| Method | | Pang et.al. | Salvetti et.al. | Beineke et.al. | Mullen & Collier | Matsumoto et.al. | Proposed approach |
|---|---|---|---|---|---|---|---|
| Naive Bayes classifier | Unigram | 81.0 | 79.5 | 65.9 | ⊗ | ⊗ | 83.65 |
| | Bigram | 77.3 | ⊗ | ⊗ | ⊗ | ⊗ | 84.06 |
| | Trigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 70.53 |
| | Unigram + Bigram | 80.6 | ⊗ | ⊗ | ⊗ | ⊗ | 86 |
| | Bigram + Trigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 83.82 |
| | Unigram + Bigram + Trigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 86.23 |
| Maximum entropy | Unigram | 80.4 | ⊗ | ⊗ | ⊗ | ⊗ | 88.48 |
| | Bigram | 77.4 | ⊗ | ⊗ | ⊗ | ⊗ | 83.22 |
| | Trigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 71.38 |
| | Unigram + Bigram | 80.8 | ⊗ | ⊗ | ⊗ | ⊗ | 88.42 |
| | Bigram + Trigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 82.94 |
| | Unigram + Bigram + Trigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 83.36 |
| Support Vector Machine | Unigram | 72.9 | ⊗ | ⊗ | 86.0 | 83.7 | 86.97 |
| | Bigram | 77.1 | ⊗ | ⊗ | ⊗ | 80.4 | 83.87 |
| | Trigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 70.16 |
| Unigram + Bigram | 82.7 | ⊗ | ⊗ | ⊗ | 84.6 | 88.88 | |
| | Bigram + Trigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 83.63 |
| | Unigram + Bigram + Trigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 88.94 |
| Stochastic Gradient Descent | Unigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 85.11 |
| | Bigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 62.36 |
| | Trigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 58.40 |
| | Unigram + Bigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 83.36 |
| | Bigram + Trigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 58.74 |
| | Unigram + Bigram + Trigram | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 83.36 |

⊗ indicate that the algorithm is not considered by the author in their respective paper

scent(SGD), and Support Vector machine (SVM). These algorithms are further applied using n-gram approach on IMDb dataset. It is observed that as the value of ‘n’ in n-gram increases the classification accuracy decreases i.e., for unigram and bigram, the result obtained using the algorithm is remarkably better; but when trigram, four-gram, five-gram classification are carried out, the value of accuracy decreases.

As discussed in Section 2.2, instead of using unigram and POS tag, the use of unigram, bigram, trigram, and their combination have shown a better result. Again, use of TF-IDF and CountVectorizer techniques as a combination for converting the text into matrix of numbers also help to obtain the value of accuracy in an improved manner, when machine learning techniques are used.

The present study has also some limitations as mentioned below:

- The Twitter comments are mostly small in size. Thus, the proposed approach may have some issues while considering these reviews.
- Different reviews or comments contain symbols like (☺, ☹, 👍, 👎) which help in presenting the sentiment, but these being images are not taken into consideration in this study for analysis.
- In order to give stress on a word, it is observed that some persons often repeat the last character of the word a number of times such as “greatttt, Fineee”. These words do not have a proper meaning; but they may be considered and further processed to identify sentiment. However, this aspect is also not considered in this paper.

In this paper, after removal of stop words, other words are considered for classification. The list of words finally obtained are observed to be very large in a good number of cases; thus in future, different feature selection mechanism may be identified to select the best features from the set of features and based on which, the classification process may be carried out. It may also happen that the accuracy value may improve, if some of the hybrid machine learning techniques are considered for classification of the sentiment. All of above mentioned limitations may be considered for the future work, in order to improve the quality of sentiment classification.

## References


Beineke, P., Hastie, T., & Vaithyanathan, S. (2004). The sentimental factor: improving review classification via human-provided information. In *Proceedings of the 42nd annual meeting on association for computational linguistics* (p. 263). Association for Computational Linguistics.

Bottou, L. (2012). Stochastic gradient descent tricks. In *Neural networks: tricks of the trade* (pp. 421–436). Springer.

Dave, K., Lawrence, S., & Pennock, D. M. (2003). Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web* (pp. 519–528). ACM.

Ekman, P., & Friesen, W. V. (1971). Constants across cultures in the face and emotion.. *Journal of Personality and Social Psychology, 17*(2), 124.

Feldman, R. (2013). Techniques and applications for sentiment analysis. *Communications of the ACM, 56*(4), 82–89.

Garreta, R., & Moncecchi, G. (2013). *Learning scikit-learn: machine learning in python*. Berlin Heidelberg: Packt Publishing Ltd.

Gautam, G., & Yadav, D. (2014). Sentiment analysis of twitter data using machine learning approaches and semantic analysis. In *Contemporary computing (IC3), 2014 seventh international conference on* (pp. 437–442). IEEE.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *Unsupervised learning*. New York: Springer.

Hsu, C. W., Chang, C. C., & Lin, C. J. (2003). *A practical guide to support vector classification*. Simon Fraser University, 8888 University Drive, Burnaby BC, Canada, V5A 1S6.

IMDb, Internet movie database sentiment analysis dataset (IMDB), 2011,

Joachims, T. (2006). Training linear svms in linear time. In *Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 217–226). ACM.

Liu, S. M., & Chen, J.-H. (2015). A multi-label classification based approach for sentiment classification. *Expert Systems with Applications, 42*(3), 1083–1093.

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies, 5*(1), 1–167.

Luo, B., Zeng, J., & Duan, J. (2016). Emotion space model for classifying opinions in stock message board. *Expert Systems with Applications, 44*, 138–146.

Matsumoto, S., Takamura, H., & Okumura, M. (2005). Sentiment classification using word sub-sequences and dependency sub-trees. In *Advances in knowledge discovery and data mining* (pp. 301–311). Berlin Heidelberg: Springer.

McCallum, A., Nigam, K., et al. (1998). A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization: 752* (pp. 41–48). Citeseer.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mouthami, K., Devi, K. N., & Bhaskaran, V. M. (2013). Sentiment analysis and classification based on textual reviews. In *Information communication and embedded systems (ICICES), 2013 international conference on* (pp. 271–276). IEEE.

Mullen, T., & Collier, N. (2004). Sentiment analysis using support vector machines with diverse information sources. In *EMNLP: 4* (pp. 412–418).

Nigam, K., Lafferty, J., & McCallum, A. (1999). Using maximum entropy for text classification. In *IJCAI-99 workshop on machine learning for information filtering: 1* (pp. 61–67).

Niu, T., Zhu, S., Pang, L., & El Saddik, A. (2016). Sentiment analysis on multi-view social data. In *Multimedia modeling* (pp. 15–27). Springer.

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* (p. 271). Association for Computational Linguistics.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10* (pp. 79–86). Association for Computational Linguistics.

Salvetti, F., Lewis, S., & Reichenbach, C. (2004). Automatic opinion polarity classification of movie. *Colorado research in linguistics, 17*, 2.

Tang, H., Tan, S., & Cheng, X. (2009). A survey on sentiment detection of reviews. *Expert Systems with Applications, 36*(7), 10760–10773.

Turney, P. D. (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* (pp. 417–424). Association for Computational Linguistics.

Zhang, D., Xu, H., Su, Z., & Xu, Y. (2015). Chinese comments sentiment classification based on word2vec and svm perf. *Expert Systems with Applications, 42*(4), 1857–1863.

Zhang, M.-L., & Zhou, Z.-H. (2007). Ml-knn: a lazy learning approach to multi-label learning. *Pattern recognition, 40*(7), 2038–2048.