SENTIMENT CLASSIFICATION USING THREE MACHINE LEARNING MODELS

Aimen Rmis[1*], Muftah Alkazagli[1], Osamah Alloush[1], Salem Almadhun[2]

[1]Department of Computer Science, Faculty of Science, Alasmarya University, Libya.
[2] Department of Computer, Faculty of Education, Elmergib University, Libya.
Corresponding author: aymenarmess2015@gmail.com

ABSTRACT

Feeling, emotions, views, and attitudes are all examples of sentiment. Because of the rapid growth of the World Wide Web, People frequently express their feelings via social media, blogs, ratings, and reviews on the internet. Due to the increase in textual data, it is necessary to examine the concept of expressing sentiments and calculate insights for business exploration. Sentiment analysis is frequently used by business owners and advertising agencies to develop new business strategies and advertising campaigns. This paper we examine the problem of document classification by sentiment. i.e. classify a document as negative document or as a positive document. We find out the machine learning algorithms (Naïve Bayes, rule based JRip and J48 trees based) preform quite efficiently on tackling this problem. We conclude by discussing more features that may make those algorithms perform even better than the results we report.

Keywords: Classification, JRip, J48 trees, Machine Learning, Naive Bayes, Sentiment analysis;

1. INTRODUCTION

Document classification has become very important in the past years with big data available everywhere that needs to be examined. One important characteristic that has been under a lot of investigation in classifying text by sentiment. With so many reviews people leave online about products, experiences, and other things, a great interest in document classification has grown. There are various reasons why people want to know what people think of their products and services. Some want to know how to improve their products, some want to know who criticizes them. With the big amount of data online, sorting those documents based on sentiment has become next to impossible [1,2].

Machine learning techniques are frequently used to identify and predict whether material is positive or negative in sentiment. Unsupervised and supervised machine learning algorithms are the two types of machine learning algorithms. The supervised method works with a labeled dataset, in which each document in the training set is assigned a sentiment. On the other hand, unsupervised method showed, with so much reviews people leave online about products, experiences and other things, a great interest in document classification has grown. There are various reasons why people want to know what people think of their products and services [3, 4].

Sentiment Analysis (SA) can be thought of as a classification technique. Document-level, sentence-level, and aspect-level SA are the three main classification levels in SA. The aim of document-level of SA is to characterize an opinion document as expressing a positive or negative mood or opinion. It views the entire document to be a single unit of basic information. Sentence-level SA seeks to categorize each sentence's sentiment, determining whether the sentence is subjective or objective as the first step. Sentence-level SA will assess whether the sentence communicates positive or negative opinions if the sentence is subjective [5].

In this paper we examine the use of popular and different machine learning algorithms on classifying text based on sentiments. The challenge with this problem comes from the characteristics that those algorithms can use to distinguish documents. People use vague indirect words to express their feeling about things. Thus, the problem of classifying based on people's feeling tend to be much harder than classifying text based on other characteristics such as classifying by topic.

The following paper is organized as follows: Section 2 presents the related work; Section 3 describes the detailed methodology of proposed algorithms; Section 4 explains the proposed approach; Section 5 shows the experimental results; Section 6 concludes the paper.

2. RELATED WORK

Vaithyanathan and Lee [1] used movie reviews data to do sentiment classification. They used data from the IMDB database. They marked reviews as positive or negatives based on the stars given to the movie that was associated with the review. They used three machine learning algorithms (Naïve Bayes, Max Entropy and Support Vector Machines) to classify three fold of 1400 reviews. They report accuracies from 77% to ~83%. In their preprocessing, they did not exclude any stop word, they did not do any kind of stemming to the word, they left the punctuations. The only noticeable thing they did were they did decomposition of conjunctions like converting the word "it's" to "it is". Turney [6]. Presents unsupervised algorithm to classify reviews as either recommended i.e., Thumbs up and not recommended. The author has used Part of Speech (POS) tagger to identify phrases that contain adjectives or adverbs. Dave et.al. [7] for testing and training, have employed organized reviews, identifying features and scoring techniques to determine if the reviews are good or bad. They utilized a classifier to categorize the sentences found through a web search query that included the product name as the search criterion. The authors[8] emphasize the importance of emotions in the process of forming consumer-brand relationships, as well as their impact on consumers' willingness to endorse a product. Emotions are generated before any processing of information, and they greatly impacted the consumer behavior. The authors

Kim et al. [9] looked at how to categorize customer sentiment based on the features of the product. Their methodology allows them to distinguish between good and negative product opinions. Unlike their forerunners, the authors have obtained a precise measurement of expressed viewpoints, this enables the determination of information relevant to a certain mobile phone device, as well as the element of the mobile phone's specification that gives it a competitive advantage over competing devices. The researchers employed a domain lexicon of 500 words with values ranging from -5 to +5 for each sentiment phrase. Other domains found that it is challenging to implement this strategy.

3. METHODOLOGY OF PROPOSED MACHINE LEARNING ALGORITHMS

Binary sentiment classification and multi-class sentiment classification are two approaches to sentiment classification that are frequently utilized in literature. Each document or review in the corpus is classified into one of two categories, positive or negative, in binary sentiment classification, each review can be divided into more than two classes in multi-class sentiment classification, such as strong positive, positive, neutral, negative, or strong negative. When two products must be compared, binary categorization is generally useful. The implementation in this study is based on binary sentiment classification [10, 11].

The goal of this study was to see if treating sentiment classification as a particular example of topic-based categorization (with the two "topics" being positive and negative sentiment) was sufficient, or whether new sentiment-categorization techniques are required. We tried out three different algorithms: Naïve Bayes classification, JRip classification, and J48

tree. The philosophies behind these three algorithms are quite different, but each has been shown to be effective in previous text categorization studies.

- Naive Bayes (NB) Classifier: It is a probabilistic classifier which uses the properties of Bayes theorem assuming the strong independence between the features. One of the advantages of this classifier is that it requires small amount of training data to calculate the parameters for prediction. Instead of calculating the complete covariance matrix, only variance of the feature is computed because of independence of features. For a given textual review 'd' and for a class 'c' (positive, negative), the conditional probability for each class given a review is P(c|d). According to Bayes theorem this quantity can be computed using the following equation [12]:

$$P(c|d) = \frac{P(d|c) * P(c)}{P(d)}$$

A further computation the term P(d|c), it is decomposed by assuming that fi's are conditionally independent from the given d's class. This decomposition of P(d|c) is expressed in following equation [13]:

$$P_{NB}(c|d) = \frac{P(c)(\prod_{i=1}^{m} P(f_i)|c)^{n_i(d)})}{P(d)}$$

- JRip Cohen, W. W., implemented JRip in 1995, and this algorithm included a propositional rule learner, Repeated Incremental Pruning to Produce Error Reduction (RIPPER). By the way, Cohen

implementing RIPPER, by altering or updating individual rules to improve rule correctness. RIPPER was used to isolate certain data for training and determine when it was time to stop adding conditions to a rule. By using the heuristic based on minimum description length as stopping criterion. Post-processing steps followed in the induction rule revising the regulations in the estimates obtained by global pruning strategy and it improves the accuracy [14].

- J48 tree, J48 supports pruning in two ways. The first is known as subtree replacement, which works by replacing nodes in a decision tree with leaf nodes. Essentially, by lowering the number of tests with a specific path. It works by starting from the leaves of a fully formed tree and working backwards to the base. The second type implemented in J48 is subtree raising, which involves moving nodes upwards toward the tree's root while also replacing other nodes. To gather the entropy gain of all these binary tests efficiently, the training data set belonging to the node in consideration is sorted for the values of the continuous attribute and the entropy gains of the binary cut based on each distinct value are calculated in one scan of the sorted data. This process is repeated for each continuous attribute [15, 16].

4. PROPOSED APPROACH

We used a corpus built from three different online review sources (Amazon.com, Yelp.com and imdb.com). Each sentence in the corpus was marked as 1 (positive) or 0 (negative). The corpus contained 3034 sentences; 1530 positive and 1504 negative. We used a python script to

extract the text and the sentiments from the original review file and wrote to one file in a Weka format. We used words and conjunction of two word (bigrams) as attributes in two different experiments setup, they were 1655 bigram attributed and 1978 unigram attributes. We exclude stop words from our data sets using the Weka stop words set. We used Weka word tokenizer to break up concatenated words into separate words when necessary. We used the default parameters in Weka for each algorithm including the smoothing values. We normalized all the values using their word frequency counts.

To run those machine learning models, we used the following standard bag of features framework. Let {f1, . . . , fm} be a predefined set of m features that can appear in a document. Let ni(d) be the number of times fi occurs in document d. Then, each document d is represented by the document vector d = (n1(d), n2(d), . . . , nm(d)) (See Fig 1).

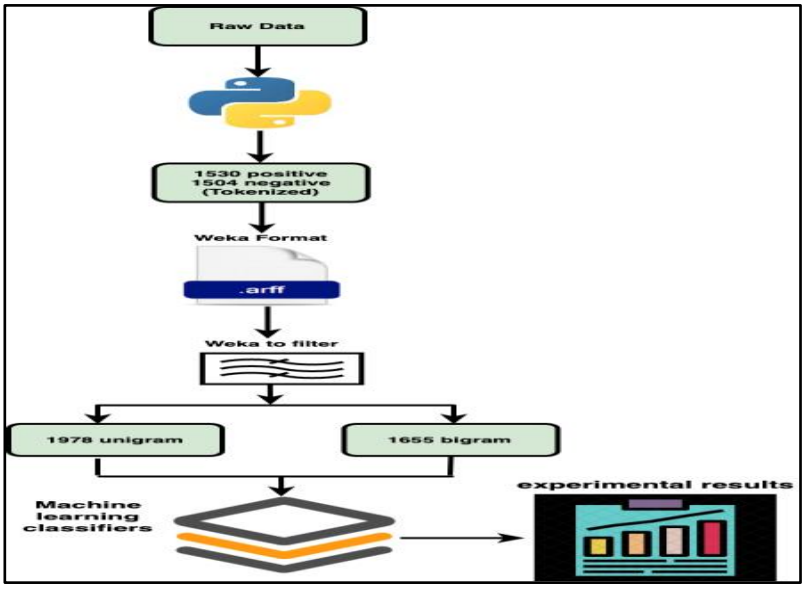

Fig 1. Diagrammatic view of the proposed approach for Sentiment Classifications

5. EXPERIMENTAL RESULTS

We start by running each model using words unigrams as a feature in 3 fold cross validation setup.

- **The Naïve Bayes** model with unigrams classified 68.9% of the sentences correctly yielding the results below.

TABLE 1: EVALUATION PARAMETERS FOR NAIVE BAYES USING WORDS UNIGRAMS.

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.779	0.401	0.664	0.779	0.717	0.766	positive
0.599	0.221	0.727	0.599	0.657	0.764	negative
Weighted Avg.	0.69	0.312	0.695	0.69	0.687	0.765

TABLE 2: CONFUSION MATRIX FOR NAIVE BAYES USING WORDS UNIGRAMS .

=== Confusion Matrix ===

a	b	<-- classified as
1192	338	a = positive
603	901	b = negative

- **The JRip rule** based model performed 62.6% correctly resulting the following values.

TABLE 3: EVALUATION PARAMETERS FOR JRIP USING WORDS UNIGRAMS.

TP Rate	FP Rate	Pre-cision	Re-Call	F-Measure	ROC Area	Class
0.522	0.267	0.666	0.522	0.585	0.667	positive
0.733	0.478	0.601	0.733	0.661	0.667	negative
Weighted Avg.	0.627	0.371	0.634	0.627	0.623	0.667

TABLE 4: CONFUSION MATRIX FOR JRIP USING WORDS UNIGRAMS.

=== Confusion Matrix ===

a	b	<-- classified as
799	731	a = positive
401	1103	b = negative

- **The J48 tree** based model classified 68.09% of the sentences correctly. Table (5) shows evolution parameters of this model.

TABLE 5: EVALUATION PARAMETERS FOR J48 TREE USING WORDS UNIGRAMS.

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.61	0.247	0.715	0.61	0.658	0.726	positive
0.753	0.39	0.655	0.753	0.701	0.726	negative
Weighted Avg.	0.68	0.318	0.685	0.681	0.679	0.726

TABLE 6: CONFUSION MATRIX FOR J48 TREE USING WORDS UNIGRAMS.

=== Confusion Matrix ===

a	b	<-- classified as
933	597	a = positive
371	1133	b = negative

We then use bigrams as features to classify documents in 3 folds cross validation setup.

- **The Naïve Bayes** model performed 69.54% with this model (better by 1% than the unigram model) Table (7) shows measures for Naïve Bayes, and table (8) shows confusion matrix.

TABLE 7: EVALUATION PARAMETERS FOR NAÏVE BAYES USING WORDS BIGRAMS.

TP Rate	FP Rate	Pre-cision	Recall	F-Measure	ROC Area	Class
0.693	0.303	0.7	0.693	0.697	0.753	positive
0.697	0.307	0.691	0.697	0.694	0.753	negative
Weighted Avg.	0.695	0.305	0.695	0.695	0.695	0.753

TABLE 8: CONFUSION MATRIX FOR NAÏVE BAYES USING WORDS BIGRAMS.

=== Confusion Matrix ===

a	b	<-- classified as
1061	469	a = positive
455	1049	b = negative

- **The JRip rule** based model using bigrams performed 61.3% correctly (1% lower than the unigrams model) resulting the following values.

TABLE 9: EVALUATION PARAMETERS FOR JRIP USING WORDS BIGRAMS.

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.473	0.243	0.665	0.473	0.552	0.657	positive
0.757	0.527	0.585	0.757	0.66	0.657	negative
Weighted Avg.	0.614	0.384	0.625	0.614	0.606	0.657

TABLE 10: CONFUSION MATRIX FOR JRIP USING WORDS BIGRAMS.

=== Confusion Matrix ===

a	b	<-- classified as
723	807	a = positive
365	1139	b = negative

- **The J48 tree** based model classified 66.24% (2% below than with unigrams) of the sentences correctly. Table (11) shows other measurements of this model.

TABLE 11: EVALUATION PARAMETERS FOR J48 TREE USING WORDS BIGRAMS.

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.647	0.322	0.672	0.647	0.659	0.716	positive
0.678	0.353	0.654	0.678	0.666	0.716	negative
Weighted Avg.	0.662	0.337	0.663	0.662	0.662	0.716

TABLE 12: CONFUSION MATRIX FOR J48 TREE USING WORDS BIGRAMS.

=== Confusion Matrix ===

a	b	<-- classified as
990	540	a = positive
484	1020	b = negative

6. CONCLUSION

The results produced by the machine learning algorithms in this paper are quite better than what we expected given the complexity of the problem. Using bigrams seems to have improved the accuracy of the Naïve Bayes model but lowered it with the rule based and trees model unlike what we have expected. We expect that the noise in this data is so high. On the other hand, unigram results the classification accuracies resulting from using only unigrams as features are showed the Naïve Bayes exhibits faster learning rate 86.9% and J48 reveals adequacy in the true positive and false positive rates 68.09%. We recommend to reduce the noise it to improve the performance is developing a larger stop words set that exclude humanly recognized sentiment words. We also expect the results get better if we can go with more n-grams features; such as thri-grams or even four-grams. The position of the words in the sentence can also be examined further; we expect have some positive impacts.

REFERENCES

[1] Pang, B., L. Lee, and S. Vaithyanathan. "*Thumbs up?: sentiment classification using machine learning techniques"*. In Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10, 2002, 79–86.

[2] S. Argamon, K. Bloom, A. Esuli, and F. Sebastiani, "*Automatically determining attitude type and force for sentiment analysis*," in Human Language Technology. Challenges of the Information Society. Springer, 2009, pp. 218–231.

[3] Y. Singh, P. K. Bhatia, and O. Sangwan, "*A review of studies on machine learning techniques,*" International Journal of Computer Science and Security, vol. 1, no. 1, 2007. pp. 70–84

[4] R. Feldman, "*Techniques and applications for sentiment analysis,*" Communications of the ACM, vol. 56, no. 4, pp. 82–89, 2013.

[5] Mikalai Tsytsarau, Themis Palpanas." *Survey on mining subjective data on the web Data Min Knowl Discov"*, 2012, pp. 478-514.

[6] P. D. Turney, *"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. Association for Computational Linguistics, 2002, pp. 417–424.

[7] K. Dave, S. Lawrence, and D. M. Pennock, *"Mining the peanut gallery: Opinion extraction and semantic classification of product reviews*," in Proceedings of the 12th international conference on World Wide Web. ACM, 2003, pp. 519–528.

[8] Hudson, S., Roth, M. S., Madden, T. J. & Hudson, R. "*The effects of social media on emotions, brand relationship quality, and word of mouth"*. An empirical study of music festival attendees. Tourism Management, 2015, 68-76.

[9] Kim, J., Choi, D., Hwang, M. & Kim, P. "*Analysis on Smartphone Related Twitter Reviews by Using Opinion Mining Techniques. Advanced Approaches to Intelligent Information and Database Systems*" , Studies in Computational Intelligence, 2014, 205-212.

[10] L. Y. Chang and W. C. Chen, "*Data mining of tree-based models to analyze freeway accident frequency*," Journal of Safety Research, 2005.36(1): 365-375.

[11] D. Delen, G. Walker, and A. Kadam, "*Predicting breast cancer survivability: A comparison of three data mining methods,"* Artificial Intelligent in Medicine, 2005.34:113- 127.

[12] S. Argamon, K. Bloom, A. Esuli, and F. Sebastiani, "*Automatically determining attitude type and force for sentiment analysis,"* in Human

Language Technology. Challenges of the Information Society.Springer, 2009, pp. 218–231.

[13] Y. Singh, P. K. Bhatia, and O. Sangwan, "*A review of studieson machine learning techniques*," International Journal of Computer Scienceand Security, vol. 1, no. 1, pp. 70–84, 2007.

[14] F. Leon, M. H. Zaharia and D. Galea, "*Performance Analysis of Categorization Algorithms,"* International Symposium on Automatic Control and Computer Science, 2004.

[15] H. Witten, and E. Frank, "*Data Mining Practical Machine Learning Tools and Techniques,*" Second Edition, Morgan Kaufmann Publisher, United States of America, 2005.

[16] Y. Zhao and Y. Zhang, "*Comparison of Decision Tree Methods for Finding Active Objects,*" National Astronomical Observatories, Advances of Space Research, 2007.