Amazon Reviews as Corpus for Sentiment Analysis using Machine Learning

Akhila Ravi[1], and Akash Raj Khettry[2] ,and Dr.Sneha Y S[3]

[1]4th year, Computer Science Engineering,JSSATEB,Bangalore,India
akhilar028@gmail.com
[2]4th year, Computer Science Engineering,JSSATEB,Bangalore,India
akashrajkhettry5@gmail.com
[3]Associate Professor,Dept of CSE,JSSATEB,Bangalore, India
sneha.girisha@gmail.com

Abstract. Most of the users today, provide their reviews on the various products on the Amazon website. The reviews provided by users are usually compact and demonstrative. For this reason, it becomes an affluent source for sentiment analysis. The objective of the paper is obtaining comparisons of the working of four standard Machine Learning algorithms for classifying the sentiments of the considered Amazon product reviews dataset. In our work, we determine the performance of these algorithms, that is, how accurately can they classify the sentiment of an unknown review. The paper provides a brief insight on sentiment analysis and the comparison of the performance of the considered algorithms of the classification of the sentiments based on several performance metrics.

Keywords: Sentiment Analysis, amazon product reviews, positive tag, negative tag, neutral tag, probability of expressions.

1 Introduction

Sentiment analysis is the process of recognizing positive and negative emotions and perspectives.. Majority of research on sentiment analysis can be seen implemented at the complete document level, for instance differentiating a negative review from the set of positive reviews. The problem of classifying documents i.e., ascertaining if a review is of a positive or a negative sentiment, is examined by considering the overall sentiment and not by the topic of the document. However, tasks such as sentiment-oriented data interpretation, and extracting product reviews needs either sentence-level or phrase-level examination of the opinion.[8] Identifying sentiments is an arduous task. The crucial problems in sentiment analysis are identifying the representation of sentiments in textual documents and determining if the expressions suggest favourable or unfavorable view toward the subject of the document.

The sentence – “Caroline’s father agreed with view of her teacher” is tagged ‘positive’ due to the presence of the positive sentiment word ‘agreed’. If the word ‘agreed’ is replaced with a negative sentiment word ‘disagreed ‘, the sentence tag would change to ‘negative’ .[5]

Sentiment analysis includes building models for two classification tasks: constructing models for classification of sentiments into positive and negative tags and a three-class classification problem of assigning positive, neutral or negative tags.[10]

Sentiment analysis involves recognition of

- Expression of sentiments,
- Probability and strength of the expressions, and
- Relationships of sentiments with the subject.

The components stated above are interrelated. For example,consider a sentence, “A conquers B”, the word “conquers” indicates a favourable opinion for A and an unfavourable opinion for B.[4]

A standard perspective to implementation of sentiment analysis proposes beginning with a set of positive and negative terms. In the considered sets, samples are labelled using their a priori prior probability. For example, the word ‘good’’ indicates a positive prior probability, and the word ‘worse’ indicates a negative prior probability.The probability in terms of context, of the whole phrase in which a particular word occurs may be quite distinct from the prior probability of the considered word.

“Ben Thomas, chancellor of the International Energy Conservation Trust, summarizes ably the propelling of the reaction of environmental campaigns and programmes: “There is no logic behind believing that the contaminators are all of a sudden, changing to become reasonable.”

In the sentence, “Trust,” “ably,” “logic,” and “reasonable” are words which have positive prior probability, but all are not included to indicate positive sentiments. The word “logic” has negative prior probability, which makes the contextual probability of the sentence as negative. The phrase “no logic behind believing” causes changes to the probability of the proposition of the sentence which follows; since “reasonable” falls is included in the considered proposition, its contextual probability is negative. The text unit “Trust” is fragment of the considered expression and it does not convey any sentiment; hence, its contextual probability is found to be neutral. Likewise the term “contaminators”: it just indicates to factories which pollute. Here, only the word “ably” has equal prior and contextual probability.

2 Background

In this section, we briefly discuss on the prior research on non topic-based text categorization. This field of work focuses on classification of the data in accordance with the style of the source or the source itself which serves as an important cue. For example, documents which involve native-language background, author, publisher (e.g., the The Daily News vs. New York Times.), and “brow” (e.g., , or low-brow or high-brow vs. “popular”) all are under this category.

Another related field of research includes ascertaining the kind of content. One of the possible categories is the ‘subjective genre’ . While techniques for categorization of genre and detection of subjectivity can aid us in recognition of documents which denote a sentiment, while not proclaiming the particular classification method of ascertaining the sentiment.[1]

Modern analysis on the classification based on sentiments have been partly based on knowledge. Few of the implemented work lays focus on classification of the semantic orientation of phrases or terms, using linguistic heuristics or a pre-selected group of seed words. The Prior work on categorization of the complete data based on sentiments, has frequently involved the utilisation of the manual or semi-manual construction of discriminant-word sets or models of cognitive linguistics.

Turney’s (2002) effort of classifying the reviews is closest to the present work. He used a particular unsupervised learning method whose basis lies on the mutual data between the words “excellent” and “poor” and the document phrases. In distinction, we use many complete supervised machine learning techniques that are free of any prior-knowledge, with the aim of analysing the innate problem of the task.

3 Proposed Approach

3.1 Feature Extraction

The dataset gathered is utilized for extracting features which are used for training the classifier. Experiments have been implemented by considering features with n-grams and for general data extraction tasks, the number of occurrence of a keyword is an appropriate feature.

The procedure of extracting n-grams from an Amazon review is:

1. Filtering – In this step we eliminate the URL links (e.g. http://amazon.com), Amazon usernames, special words and characters.
2. Tokenization – Text is divided and tokenized by using spaces as a delimiter and removing punctuation symbols to construct a bag of words. Stemming and Lemmatization are used for further text processing.
3. Removing stop words– From the processed text, we exclude stop

words-“the”, “a” and “an” from the constructed set of words.
4. Constructing n-grams–We construct a corpus of n-grams from a set of continuous terms. A negation word like “not” and “no” is usually connected to a term which is prior to the term or follows it. Consider an instance, “I will not play ball” results in formation of two bigrams: “I will+not”, “will+not play”, “not+playball”.

3.2 Naive Bayes Approach

One approach for classifying textual documents is to tag a considered record d of a class c ‘ = argmax$_c$ P(c | d). Using Bayes’ rule, derivation of the equation for the Naive Bayes classifier model. The Bayes’ rule is observed as,

$$P(c \mid d) = \frac{P(c)P(d \mid c)}{P(d)} \qquad (1)$$

where P(d) has no contribution in selecting c ‘ . For estimating P(d | c),given class of d, the classifier fragments it by presuming the term fi ’s are conditionally independent. The proposed training function comprises of frequency estimation of P(c) and P(fi | c), by utilising smoothing.[1]

3.3 Support Vector Machine Approach

Support vector machines (SVMs) are generally more effectual at standard text categorization, usually surpassing Naive Bayes. Rather than being probabilistic classifiers,these classifiers are large-margin classifiers ,in distinction to Naive Bayes. For the bi-class case, the general intent backing the training method is identifying a hyperplane, denoted as vector ŵ, which not just distinguishes the records vectors in a class from another, but also ensures large margin of separation. The study correlates to a constrained optimization task. Let $c_j \in \{1, -1\}$ (denoting positive and negative respectively) be assumed as the right class for document d_j , then the equation is represented as

$$\vec{w} := \sum_j \alpha_j c_j \vec{d}_j, \quad \alpha_j \geq 0, \tag{2}$$

where the α_j 's can be retrieved by finding the solution for a problem which is dual optimization. [1]

3.4 Random Forest Classifier Approach

The notion of ensembling decision trees is called as Random Forest, which can be obtained through integration of several decision trees. On considering single tree classifiers such as decision tree classifier, we may encounter issues such as outlier data or noisy data, which can influence the performance of the classifier function, whereas Random Forest as a classifier provides randomness and hence it is highly robust to noise and outliers. This classifier produces two kinds of randomness, one relating to data randomness and the other relating to features randomness. Since this classifier deals with integrating several Decision Trees, it includes many hyperparameters such as:

- How many trees are to be constructed for the Decision Forest.
- How many features are to be selected randomly.
- The depth of every tree.

Random Forest is regarded as an accurate and robust classifier since it involves the concept of bootstrapping and bagging. [12]

3.5 K Nearest Neighbor Approach

KNN classifier is a type of instance based learning. In this approach, the classifier function is approximately local and all the computations are carried over until classification. Of all the machine learning algorithms,this is one of the simplest approach. In KNN classification, the output is class label to which a particular instance belongs to. An instance is classified by maximum scores of its neighbours with the instance being assigned to the class which has more similarity among its k nearest neighbour (here k denotes a small, positive integer). The nearest neighbour is determined using similarity index measures; generally distance functions are used. The distance functions used commonly by KNN are [13].

The Euclidean distance function is computed as

$$dist(A,B) = \sqrt{\frac{\sum_{i=1}^{m}(x_i - y_i)^2}{m}} \quad (3)$$

Manhattan distance function is computed as

$$D_{MANHATTAN}(x_i, x_j) = \sum_{k=1}^{d} \left| x_{ik} - x_{jk} \right| \quad (4)$$

4 Data and Methodology

4.1 Data Corpus

The classification models are tested on an actual corpus of Amazon product reviews. In a time-frame of over two decades, several customers have contributed reviews to express their view and experiences regarding the purchased products. Each review can be analyzed to have one among the three sentiments- positive, negative or neutral.

The dataset is the standard dataset available in the ‘Kaggle’ website.The dataset obtained was raw and contained many redundant samples.After pre-processing of data,it was reduced to 119 distinct samples.The distribution of the Amazon product reviews dataset are presented in Table 1

Table 1. Distribution of instances in the dataset.

Sentiment	Number of Samples
Positive	65
Negative	33
Neutral	21
Total	119

4.2 Classifier

We built a sentiment analyzer classifier using four algorithms : Multinomial Naıve Bayes classifier, Support vector machine,Random Forest Classifier and KNN classifier. However, among these three classifiers the Random Forest Classifier and Naive Bayes classifier yielded the best results.

Performance metrics are used to obtain comparison of different classifier models implemented. We compute accuracy of the sentiment analysis classifier on the complete evaluation dataset, as

$$accuracy = \frac{N(\text{correct classifications})}{N(\text{all classifications})} \tag{5}$$

5 Results

The four algorithms,Naive Bayes algorithm, Support Vector Machine algorithm, Random Forest Algorithm and K-Nearest Neighbor algorithm, are implemented on the considered corpus and the following observations are obtained.

The processed dataset was further split into training and testing sets of varied sizes. The different sized training and testing sets were implemented for the complete set of four algorithms and the accuracy obtained is as

Table 2. Comparison of four algorithms for different sizes of training and testing sets.

Train size	Test size	Naive Bayes	Support Vector Machine	Random Forest	K-Nearest Neighbor
67	33	65	55.95	60.71	61.9
40	60	63.88	58.33	61.11	68.05
50	50	66.66	50	70	60
60	40	60.41	52.08	68.75	56.25

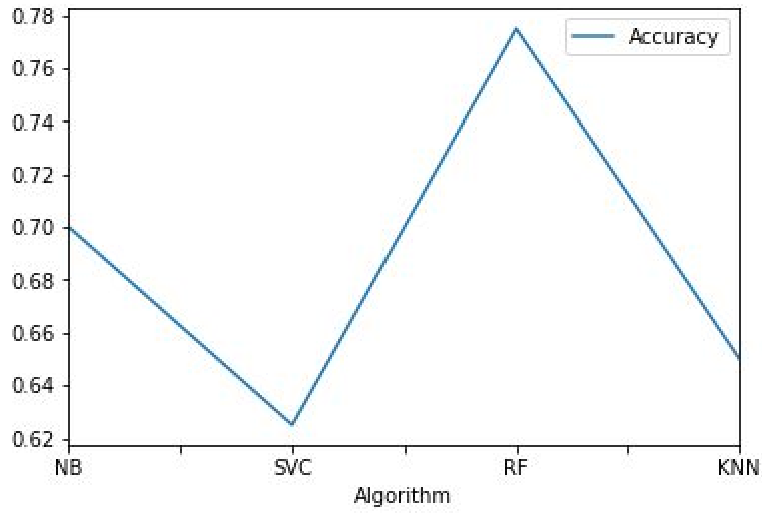

Fig. 1. Performance of the algorithms based on the accuracy

The above graph depicts accuracy produced by the four Machine Learning algorithms considered. We observe that the Random Forest classifier provides highest accuracy for the given dataset, while support vector machine provides the least accuracy.

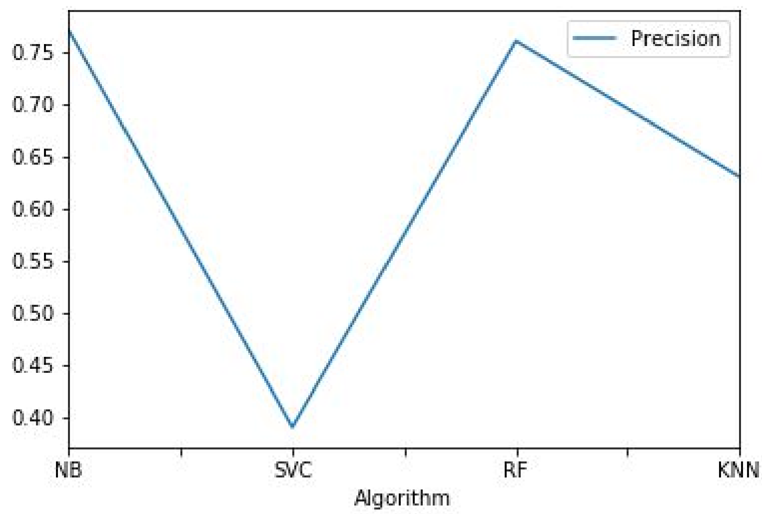

Fig. 2. Performance of the algorithms based on the precision

The above graph depicts precision of the four Machine Learning algorithms considered. We observe that the Naive Bayes classifier provides highest precision for the given dataset, while support vector machine provides the least precision.

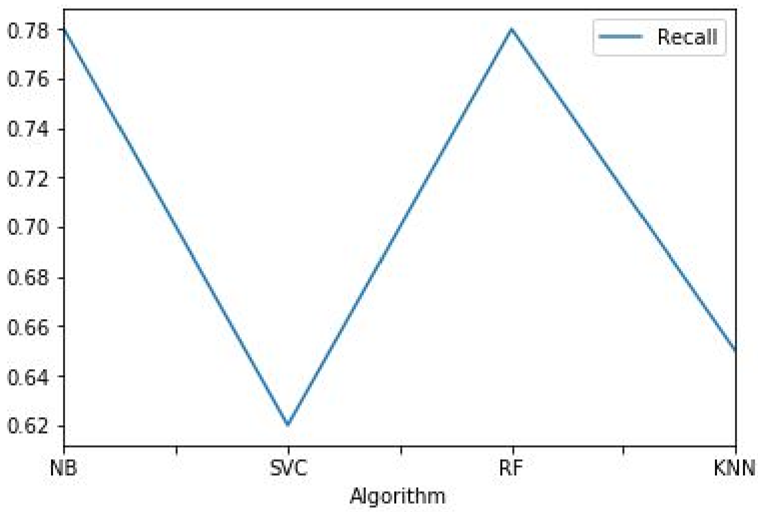

Fig. 3. Performance of the algorithms based on the recall

The above graph depicts recall of the four Machine Learning algorithms considered. We observe that the Naive Bayes classifier and Random Forest classifier provides highest recall for the given dataset, while support vector machine provides the least recall.

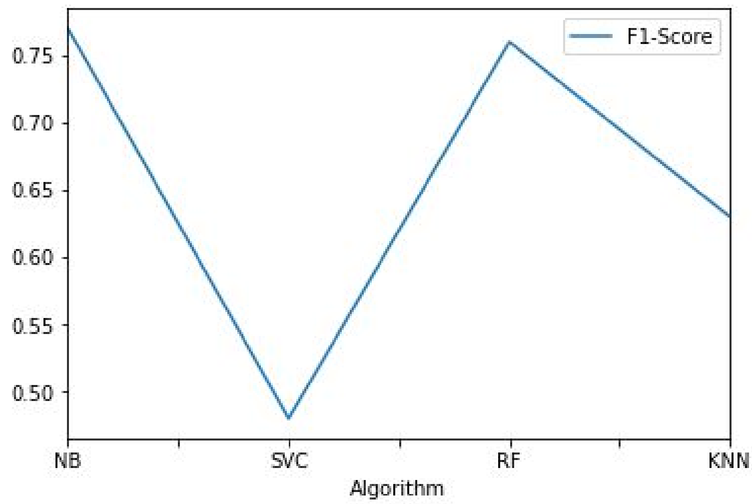

Fig. 3. Performance of the algorithms based on the F1-Score

The above graph depicts F1-Score of the four Machine Learning algorithms considered. We observe that the Naive Bayes classifier provides highest F1-Score for the given dataset, while support vector machine provides the least F1-Score.

Table 3. Comparison of four algorithms with performance metrics.

Algorithm	Accuracy	Precision	Recall	F1-Score
Naive Bayes	70	0.77	0.78	0.77
Support Vector Machine	62.5	0.39	0.62	0.48
Random Forest Classifier	77.5	0.76	0.78	0.76
K-Nearest Neighbor	65	0.63	0.65	0.63

6 Conclusion

In this paper, a comparison of Machine Learning Algorithms for sentiment analysis of Amazon product reviews is demonstrated. The large amount of data available as Amazon review dataset makes it a striking source of data for sentiment analysis and classification of sentiment of the review. We have exemplified a sentiment analysis approach using Machine Learning algorithms for extricating sentiments corresponding to the positive or negative polarities for reviews in a text-document. However we could represent sentiments using varied interpretations which includes indirect interpretations which requires a certain amount of analytical reasoning, for a sentiment to be identified. Hence it has been challenging to show the expediency of our elementary scheme of sentiment analysis. The experimental observations obtained, however, demonstrate that insightful data on sentiments from almost all the documents, can be extracted using the proposed models.Out of the four implemented models, the Random Forest Classifier model produced the best accuracy metrics. In future work, we intend to explore even richer and sophisticated methods of linguistic analysis like, parsing, semantic analysis and topic modeling. The classifier models are capable of determining positive, negative and neutral sentiments of documents.

References

1. Bo Pang, Lillian Lee, Shivakumar Vaithyanathan.:Thumbs up? Sentiment Classification using Machine Learning Techniques(2002)
2. Kushal Dave, Steve Lawrence, David M Pennock.: Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews(2003)
3. Jeonghee Yi, Tetsuya Nasukawa, Razvan Bunescu, Wayne Niblack.:Sentiment Analyzer: Extracting Sentiments about a Given Topic using Natural Language Processing Techniques(2003)
4. Tetsuya Nasukawa, Jeonghee Yi.:Sentiment Analysis --Capturing favorability using Natural Language Processing--(2003)
5. Soo-Min Kim, Eduard Hovy.:Determining the Sentiment of Opinions(2004)
6. Philip Beineke,Trevor Hastie,Shivakumar Vaithyanathan.:The Sentimental Factor: Improving Review Classification via Human-Provided Information(2004)
7. Michael Gamon.:Sentiment classification on customer feedback data:noisy data,large feature vectors, and the role of linguistic analysis(2004)
8. Theresa Wilson,Janyce Wiebe,Paul Hoffman.:Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis(2005)
9. Alexander Pak,Patrick Paroubek.:Twitter as a Corpus for Sentiment Analysis and Opinion Mining(2010)
10. Apoorv Agarwal,Boyi Xie,Ilia Vovsha,Owen rambow,Rebecca Passonneau.:Sentiment Analysis of Twitter Data(2011)
11. Parmar,Hitesh and Bhanderi,Sanjay & Shaw,Glory.:Sentiment Mining of Movie Reviews using Random Forest with Tuned Hyperparameters(2014)
12. A.M.Abirami,V.Gayathri.:A Survey On Sentiment Analysis Methods and Approach(2016)
13. Shubam Goyal.:Review Paper on Sentiment Analysis of Twitter Data Using Text Mining and Hybrid Classification Approach(2017)