Journal of Information Technology and Digital World (ISSN: 2582-418X)
www.irojournals.com/itdw/

Analogy of Machine Learning Approaches and BERT for Sentiment Analysis

K. Vidya[1], S. Janani[2]

Department of Computer Science and Engineering, St. Joseph's College of Engineering, Chennai, India

E-mail: [1]vidya.visahan@gmail.com, [2]sjanani.me@gmail.com

Abstract

For assessing customer sentiment in Amazon product reviews, this article compares two machine learning algorithms and a deep learning method, BERT (Bidirectional Encoder Representations from Transformer). Machine learning is the most practical approach in the current era of artificial intelligence for training a neural network to handle the majority of real-world issues. In this paper, the real-world scenario of sentiment analysis is considered, ideally the classification problem. Firstly, the data is provided into a model, which evaluates the feature that uses the Term Frequency (TF) and Inverse Document Frequency (IDF) pre-processing methods. Secondly, the algorithms, Naive Bayes classifier and Support Vector Machine are used to analyze the sentiment of the consumer comments and compute metrics like F1 score. Finally, the input data is fed for BERT to process and compute the F1 score. To summarize, this study is to provide a detailed comparative analysis of machine learning techniques and deep learning algorithms.

Keywords: Sentiment analysis, Naïve Bayes classifier, SVM, BERT

1. Introduction

Rather than making purchases at marketplaces and stores, people currently choose to trade through e-commerce websites. People rely heavily on product reviews and comments when purchasing things online, and these reviews and comments have a significant effect on people's purchasing decisions. However, skimming through hundreds of review comments takes time, and it seems cruel to do so. This is where machine learning algorithms enter the fray to aid us in solving our real-time problems. In this case, an assessment of a certain category review may be polarized to evaluate its worldwide appeal. Sentiment Analysis (SA) employs logic to extract a user's thoughts and emotions. It's a text classification system that

DOI: https://doi.org/10.36548/jitdw.2022.1.006
Received: 13.04.2022, received in revised form: 28.04.2022, accepted: 16.05.2022, published: 26.05.2022

categorizes texts according to the sentiment orientation of the opinions they include. As a result, it is crucial in Natural Language Processing (NLP). NLP is a discipline of computer science and artificial intelligence that studies the interplay of human and machine language.

Merchants, stock traders, and election workers all benefit from this sector. The method of recognizing the text's contextual polarity is known as sentiment analysis. It decides whether a given text is positive, negative, or neutral [2]. It's also known as opinion mining since it derives the speaker's viewpoint or attitude. Figure 1 depicts the overall system architecture.

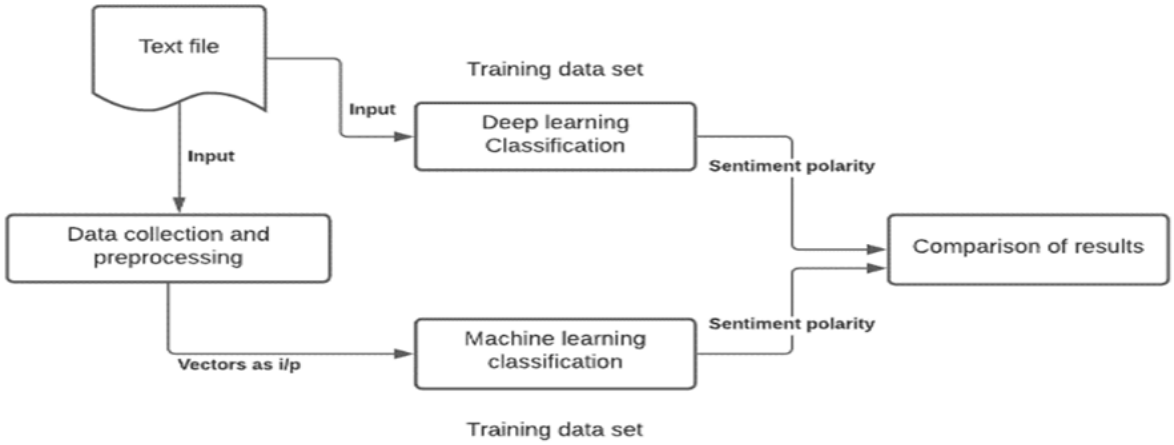

Figure 1. System architecture's flow diagram

To conduct the research for this study, product reviews from an e-commerce website have been used. The Naive Bayes classifier [7], Support Vector Machine [8] (both machine learning algorithms), and BERT [4] (a deep learning algorithm) are utilised for comparison. The first step toward implementation is the data pre-processing approach, for which the TF-IDF (Term Frequency - Inverse Document Frequency) algorithm is applied. The input text file is vectorized as part of the pre-processing procedure. Both of our machine learning algorithms take the result of the pre-processing stage as input [6]. Because BERT includes an in-built pre-processor that can handle text input, it utilises the text file directly as input [5].

2. Related Work

There are so many papers that research the concepts of Sentiment analysis using various algorithms. In the work [2] done by Tanjim Ul Haque; Nudrat Nawal Saber; Faisal Muhammad Shah, they have used machine learning algorithms like Naïve Bayesian, SVM, Stochastic Gradient Descent (SGD), Linear Regression (LR), Random Forest and Decision

Tree. They conducted experiments with different feature selection techniques like TF-IDF and bag of words. The accuracy of all the algorithms is compared along with the other statistical measurements.

The work done by [1] Sanjay Dey et al., used three different feature extraction methods namely TF-IDF, Frequent noun identifier and relevant noun removal. Two machine learning algorithms Naïve Bayes and Linear SVM were used for the comparative study. The study provided with a result where the SVM performs better than Naïve Bayes.

The link mentioned in [3] is where the dataset for this paper has been obtained. This Kaggle website provides with the required e-commerce product review dataset. The work [4] done by Jacob Devlin et al., is the first paper developed for a new language representation model called BERT. This is the base paper which explains the working of the BERT algorithm. On eleven natural language processing tasks, it achieved new state-of-the-art outcomes.

The work [5] done by Zhengjie Gao et al., presents with the new modifications in the BERT algorithm. It introduces two improvised concepts of BERT, i.e., TD-BERT and BERT-FC. A comparison has been provided among these algorithms with other NLP techniques as baselines. In work [6] done by Shweta Rana and Archana Singh, the authors provided a comparative study of the movie review dataset using SVM and Naïve Bayes Classifier. They have used the Porter algorithm to achieve better results for text processing.

In work [7] done by Palani Thanaraj Krishnan, Alex Noel Joseph Raj, and Vijayarajan Rajangam the authors worked on speech recognition using SVM. The link in [9] provides extensive knowledge about the TF-IDF algorithm and its library features. This way how the algorithms work on various levels can be understood extensively. The improved TF-IDF algorithm was provided in the work done by [10] Cai-zhi Liu et al. The improved algorithm addressed the problem of ignoring contextual semantic links thus providing better features.

3. Methodology

Amazon is among the most popular e-commerce sites, as seen by the numerous reviews available. The dataset was unlabeled, and to utilize it in a supervised learning model, it needs to be labelled. Finally, this study activity was limited to Amazon product feedback, namely, book comments. For measuring polarization, about 1,47,000 book reviews were examined.

3.1 Data Collection

The initial phase in the process should ideally be data collection. The dataset obtained from the Kaggle website has around 1.50 lakh entries [3]. The dataset is not labelled, thus before applying the TF-IDF method to pre-process it, it must be labelled using approaches such as an active learner.

3.2 Data Pre-processing

Data pre-processing involves various data cleaning steps. The techniques used in this study are as follows:

1. Change the text to lower cases:

Because the machine treats lower and upper cases differently, it is easy for a computer to read the words if the text is in the same case. To avoid the differing perceptions of terms, the content should be written in the same case, with the lower case being the preferred option.

2. Remove stop words:

Stopwords are the words that appear repeatedly in a text yet provide no meaningful information. Stopwords include words like they, this and where, among others. With around 180 stopwords eliminated, the NLTK library is an extensively used library for eliminating stopwords. Any new term may be readily added to a list of words.

3. Removing categorization labels:

Label 1 and label 2 in the dataset represented classification for all of the items. Customization has been implemented to delete this label prefix from the entire dataset because it isn't beneficial for this study.

4. TF-IDF Implementation:

The TF-IDF retrieval approach takes into account both the frequency of a phrase (TF) and the inverse frequency of documents (IDF) [9]. Each term or phrase has a TF and IDF score. Meanwhile, a phrase's TF and IDF product outcomes are tied to the term's TF-IDF weight. As an outcome, the higher the TFIDF score, the rarer the word, and vice versa (weight). Consequently, a word's TF shows its frequency, and the IDF represents how significant that phrase is across the corpus. If the document's phrases have a high content TF-IDF weight, the content will always rank in the top search results, allowing anybody to

prevent stopwords while simultaneously identifying words with higher search intensity and lower competition. [10].

Algorithm of the system

1. Start
2. Data set is collected from the open-source website known as Kaggle.
3. The data now is pre-processed using a TF-IDF pre-processor to convert the text into vectors.
4. The pre-processors output is fed as input to the ML algorithm.
5. The models analyze the input and provide the F1 score.
6. The text file is simultaneously fed as input to the BERT model.
7. The model analyses the input and provides the F1 score.
8. The statistical measures of Naïve Bayes, SVM and BERT are thus compared.

4. Experimental Results

This section assesses the performance of these two machine learning models and one deep learning model through a series of experiments. Initially, the dataset is analyzed and preprocessed for further work. The dataset is vectorized using the TF-IDF algorithm, which is the data pre-processing stage and the output of this stage is given as input for both ML approaches. The dataset is split for training and testing purposes. Both Naïve Bayes and SVM algorithms take in the vectorized data and provide us with required statistical measurements. On the other hand, BERT takes in the entire data set without preprocessing it. BERT has its own inbuilt preprocessing techniques. Few steps are involved after processing the dataset. Firstly, the dataset is split for training and testing purposes Secondly, the dataset must be tokenized and the data with those tokens are encoded. Thirdly, the pre-trained BERT model is used and data loaders, optimizers and schedulers are created. Finally, the model is trained for the training dataset and used for verifying it against the test dataset. The batch size for each run is made 16 and the minimum number of epochs which is 10 is obtained.

Evaluating metrics is important for determining classification efficiency, and assessing accuracy is the easiest way to do so. Ultimately, the accuracy of a classifier on a given test dataset is the fraction of those datasets that it correctly categorizes. The system is evaluated using three extensively used statistical measures: recall, precision, and the F-measure, which is derived from a confusion matrix. The confusion matrix is a table that illustrates how well a classification model performs on a set of test data only when actual data

is provided. The nomenclature associated with the confusion matrix might be perplexing. The most fundamental terms used in a confusion matrix are as follows:

True Positive: Predicted values were accurately predicted as positive.

False Positive: Anticipated values predicted a real positive wrongly. In other words, negative values are projected to be positive.

False Negative: Positive values are projected to be negative.

True Negative: Predicted values that are accurately predicted as negative.

Therefore, Table 1 and Table 2 provide statistical metrics of Nave Bayes, SVM, and BERT, respectively, for comparison.

Table 1. Statistical Measurement of Naive Bayes, SVM and BERT

Algorithm	**Statistical Measurements**		
	Precision	**Recall**	**F1-Score**
Naive Bayes	0.81	0.86	0.83
SVM	0.85	0.86	0.86
BERT	0.94	0.96	0.95

Figure 2 provides a graphical representation of the statistical measurements of all three algorithms.

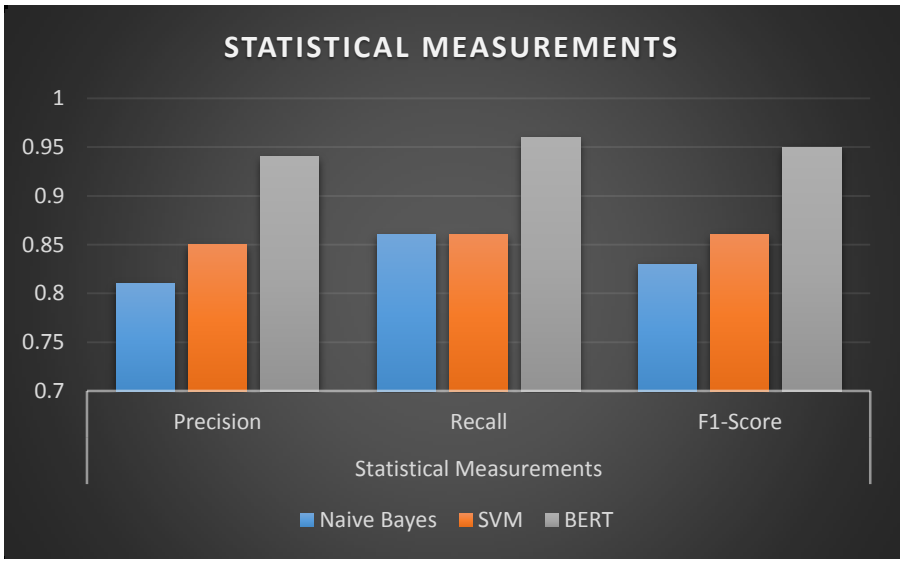

Figure 2. Graphical representation of the statistical measurements

Precision is a ratio of all positive instances. It is denoted by the below equation (1),

$$\text{Precision} = TP/\ TP+FP \quad (1)$$

The recall refers to the classifier's capacity to locate all positive samples. It is denoted by the equation below (2),

$$\text{Recall} = TP/TP+FN \quad (2)$$

The F1 score is the metrics' industry norm. It's a weighted average of precision and recall, with 1 being the highest and 0 being the lowest. It is denoted by the equation below (3).

$$\text{F1 Score} = 2*\text{Recall}*\text{Precision}/\ \text{Recall} + \text{Precision} \quad (3)$$

The F1 score for all the 10 Epochs is given below in Table 2.

Table 2. F1 Score of Every Epoch for BERT

Epoch #	**F1 Score**
1	0.9000
2	0.9352
3	0.9407
4	0.9469
5	0.9390
6	0.9454
7	0.9469
8	0.9461
9	0.9468
10	0.9477

Figure 3 provides a graphical representation of the F1 scores for 10 epochs.

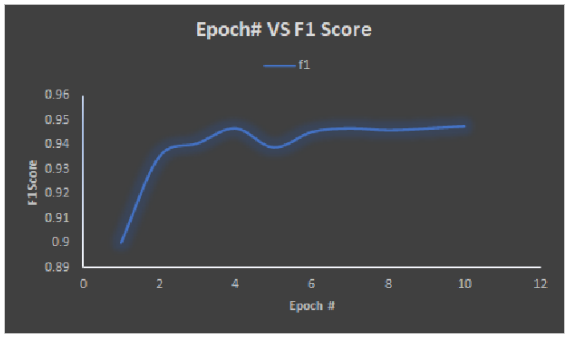

Figure 3. Graphical representation of F1 score for BERT

5. Conclusion & Future Scope

This study was able to compare the Naive Bayes classifier, SVM and BERT, to examine the polarization of Amazon product evaluations. After the pre-processing step, the models are trained using over 2250 features and almost 6000 datasets. The Naïve Bayes classifier in this system has an accuracy of 83.54 percent, a precision of 0.81, a recall of 0.81, and an f1 score of 0.83. The data are also categorized with the SVM, which had an accuracy of 85.49 percent, a precision of 0.85, a recall of 0.86 and an f1 score of 0.86. BERT has a precision of 0.94, a recall of 0.96 and an f1 score of 0.95. Comparing the obtained results with paper [1], the ML approaches provide a higher F1 score value, and for the same dataset the deep learning algorithm BERT provides the best F1 score of 0.95 in this model. After comparing the statistical measurements, it is found that BERT provides a higher F1 score for the dataset. Thus, the BERT model is better when compared to the ML approaches.

Future enhancement for this study can be extended by including neutral statements for classification, as only positive and negative classification has been used currently. This model can be extended for multiple emotions like happy, sad, sarcasm, anger, surprise, disgust, etc.

References

[1] S. Dey, S. Wasif, D. S. Tonmoy, S. Sultana, J. Sarkar, and M. Dey, "A comparative study of support vector machine and naive Bayes classifier for sentiment analysis on Amazon product reviews," in 2020 International Conference on Contemporary Computing and Applications (IC3A), 2020, pp. 217–220.

[2] T. U. Haque, N. N. Saber, and F. M. Shah, "Sentiment analysis on large scale Amazon product reviews," in 2018 IEEE International Conference on Innovative Research and Development (ICIRD), 2018, pp. 1–6.

[3] Prithvi, "Sentiment analysis with Amazon Reviews," Kaggle.com, 23-Jan-2021. [Online]. Available: https://www.kaggle.com/code/prithvi57/sentiment-analysis-with-amazon-reviews/data.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional Transformers for language understanding," arXiv [cs.CL], 2018.

[5] Z. Gao, A. Feng, X. Song, and X. Wu, "Target-dependent sentiment classification with BERT," IEEE Access, vol. 7, pp. 154290–154299, undefined 2019.

[6] S. Rana and A. Singh, “Comparative analysis of sentiment orientation using SVM and Naive Bayes techniques,” in 2016 2nd International Conference on Next Generation Computing Technologies (NGCT), 2016, pp. 106–111.

[7] P. T. Krishnan, A. N. Joseph Raj, and V. Rajangam, “Emotion classification from speech signal based on empirical mode decomposition and non-linear features: Speech emotion recognition,” Complex intell. syst., vol. 7, no. 4, pp. 1919–1934, 2021.

[8] V. G. V. Mahesh, C. Chen, V. Rajangam, A. N. J. Raj, and P. T. Krishnan, “Shape and texture aware facial expression recognition using spatial pyramid Zernike moments and law’s textures feature set,” IEEE Access, vol. 9, pp. 52509–52522, undefined 2021.

[9] “Sklearn.Feature_extraction.Text.TfidfTransformer,” scikit-learn. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.Tfid fTransforme r.html.

[10] C.-Z. Liu, Y.-X. Sheng, Z.-Q. Wei, and Y.-Q. Yang, “Research of text classification based on improved TF-IDF algorithm,” in 2018 IEEE International Conference of Intelligent Robotic and Control Engineering (IRCE), 2018, pp. 218–222.

[11] Y. Xu, X. Wu, and Q. Wang, “Sentiment analysis of yelp’s ratings based on text reviews,” Stanford.edu. [Online]. Available: http://cs229.stanford.edu/p roj2014/Yun%20Xu,%20Xinhui%20Wu,%20Qinxia%20Wang,%20Sentiment%20Anal ysis%20of%20Yelp's%20Ratings%20Based%20on%20Text%20Reviews.pdf.

[12] C. Rain, “Sentiment analysis in Amazon reviews using probabilistic machine learning,” Swarthmore.edu. [Online]. Available: https://www.sccs.swarthmore.edu/users/15/ crain1/files/NLP_Final_Project.pdf.

[13] Bhatt, A. Patel #, H. Chheda #, and K. Gawande, “Amazon Review Classification and Sentiment Analysis,” Psu.edu. [Online]. Available: http://citeseerx.ist.psu.edu/viewd oc/download?doi=10.1.1.736.4819&rep=rep1&type=pdf.

[14] W. Chen, C. Lin, and Y.-S. Tai, “Text-based rating predictions on Amazon Health & Personal Care product review.”

[15] R. Xia, F. Xu, C. Zong, Q. Li, Y. Qi, and T. Li, “Dual sentiment analysis: Considering two sides of one review,” IEEE Trans. Knowl. Data Eng., vol. 27, no. 8, pp. 2120–2133, 2015.