Aspect Based Sentiment Analysis of Customer Reviews Using Deep Learning Algorithm

Ms. M.P. Geetha[1], Yashodhaa.N[2], Perasanna.S[3], Prideesh Kumar.M[4]
[1]Assistant Professor, Dept of CSE
[2, 3, 4]Dept of CSE
[1, 2, 3, 4] Sri Ramakrishna Institute of Technology

Abstract- *Nowadays consumers post reviews on many E-commerce websites to share their thoughts and opinions. These websites have many reviews of several commodities, products, and other services. With these data Sentiment Analysis can be performed which classifies the sentiments conveyed in the written review and can be evaluated as positive, negative, or neutral. The goal of Sentiment Analysis is to extract subjective information from text and provide insights into the customer's opinions and attitudes towards a product or service. Sentiment Analysis can be done using various techniques, such as rule-based methods, machine learning algorithms, and deep learning models. These techniques analyse various aspects of the text, such as the words used, the context in which they appear, and the sentiment expressed by phrases and sentences. Overall, Sentiment Analysis of reviews is a valuable tool for businesses looking to improve customer satisfaction, brand reputation, and overall business performance.*

Keywords- Sentiment Analysis; BERT; Multinomial Naive Bayes; Deep Learning; Customer reviews; Text Analysis

I. INTRODUCTION

Internet evaluations are a crucial source of information for customers and may greatly affect a company's success and reputation. It's important for consumers to read online reviews as a source of information and they must exercise caution when evaluating them. The businesses should actively monitor and manage these online reviews about their business to ensure they are accurately representing their products or services. Moreover, as the amount of reviews is rapidly growing, people face a serious problem of analysing the right emotions of the customers and to know how a customer really feels about the product. A major issue in performing Sentiment Analysis is analyzing and classifying the polarity(positive, negative and neutral emotions) of a given text at the document, sentence, or feature or aspect level. So, it is difficult for majority of the available tools to precisely evaluate what's truly is negative, neutral and a positive review and achieve high Sentiment Analysis accuracy. However, analysing customer reviews to identify their sentiments has been proven to be challenging and time-consuming due to high volumes of review data collected from various review sources.

So, the approach of Aspect-Based Sentiment Analysis (ABSA) can be used to classify the sentiments of various customers on each product and services. Aspect based Sentiment Analysis is a text classification technique that can categorize the review data and identifies the emotions that are attributed to each data. It is used to analyse customer's feedback by identifying the right emotions with different aspects or features of a product or a service. Here the aspects means the features of a product or a service. ABSA can extract sentiments and aspects. The sentiments here refers the positive, negative or neutral reactions about a specific aspect and the aspect means the topic that is being talked about. Therefore, ABSA is an important key to automatically analyse the customer's review or opinions for taking data-based decisions and to know overall feedback of a product.

II. LITERATURE SURVEY

1) **Seungwan Seo, Czangyeob Kim, Haedong Kim, Kyounghyun, Pilsung Kang, "Comparative Study of Deep Learning Based Sentiment Classification", [2020].**

In this study, the CNN-based model and RNN-based model is compared for Sentiment Classification. Increasing the number of RNN layers did not guarantee a performance improvement. Between the model structures, if all the conditions are made same, then the RNN-based model can outperform the CNN-based model. In contrast to the training dataset volume, a consistent vocabulary size for both RNN and CNN-based models does not exist. Based on the depth of the convolution layers, the CNN model reacted differently to the input level. Here the character level input is significantly higher than the word level input. Thus, increasing the layers can help the model in understanding the long-range relationship within the text better. This will result in improved performance. For the CNN models, it cannot be concluded that increasing the model complexity will improve or impair the performance of classification. From this paper it can be said that increasing the model complexity will always improve the performance of the model in case of RNN models.

2) Alhassan Mabrouk, Rebeca P, Diaz Redondo, "Deep Learning Based Sentiment Classification: A Comparative Survey", [2020].

Lately, Deep Learning (DL) models are used to solve the problems in classification of sentiments, which is a core concept in Sentiment Analysis (SA). The performances of these models that are mentioned in this paper are affected by different factors. This literature survey is about a comprehensive survey of the state-of-the-art Deep Learning methods for Sentiment Classification. SA has emerged with the aim of analyzing and categorizing user opinions. Sentiment Analysis is a relevant for different fields, like: sales measurement, analyzing brand reputation, business competition analysis, market research, customer service feedback etc. In fact, both individual persons and organizations are interested in SA. In Sentiment Classification, although many approaches have been proposed on different levels of SC: aspect, sentence and document, this paper provides a comparative and a comprehensive literature-based survey for the existing DL approaches. The three most commonly used Neural Network architectures are RNN, CNN and Hybrid NN and these are used to solve the Sentiment Classification on three levels of document, sentence and aspect respectively. These perform well with the document-level Sentiment Classification. After evaluating all the models, the RNN model achieved the highest results. The performance of DL-based models on the aspect level Sentiment Analysis is still a challenge and needs much more efforts to be solved.

III. METHODOLOGY

Deep learning is a powerful approach for Sentiment Analysis, as it can learn representations of text data that capture complex patterns and relationships between words. Deep learning models are particularly well-suited for the process of Sentiment Analysis because they can learn from large amounts of data and capture the nuanced relationships between words and their context. Therefore, a deep learning model is used to classify the emotions in the given sentence. It is used to determine the attitude of the mass towards the subject of interest or the aspect. The following are the steps to perform sentiment analysis. Here, two algorithms are used to perform sentiment analysis on customer reviews.

A. Dataset

A suitable dataset is required to perform the sentiment or the emotion of the customer in their reviews. Dataset formats differ according to different use cases. These data are uncleaned raw data which should be cleaned and pre-processed before classifying the sentiments. For this project we use Google Play Store Reviews.

B. Data Quality Assessment

(i) *Mismatched data types:* When data is collected from different locations or sources, it may come in different formats. Thus, all the mismatched data must be reformatted or replaced.
(ii) *Mixed data values:* Different sources use different values for features. These values should be made uniform.
(iii) *Missing data:* There will be missing data fields, blank spaces in text, or unanswered fields in the dataset that is collected. This may occur due to human negligence. To manage these missing data, data cleaning is an essential process.

C. Data Cleaning

Data cleaning is one of the most essential steps in data pre-processing because this process will ensure that the data that is collected by us is suitable for the model to process. In short data cleaning is the process of correcting, adding, repairing, or removing irrelevant data from the dataset that is collected. Data cleaning will rectify all the inconsistent data that are uncovered in data quality assessment.

D. Multinomial Naïve Bayes Approach (MNB)

Multinomial Naive Bayes is a type of Naive Bayes algorithm that is commonly used for text classification problems. It is based on the Bayes theorem, which calculates the probability of a hypothesis based on prior knowledge of conditions that might be related to the hypothesis. Naïve Bayes formula:

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

Where,

P(A|B) - measure of how often A and B are observed to occur together (posterior probability)
P(B|A) - measures of how often B occur in A (likelihood)
P(A) - measure of how often A is observed to occur in general (prior probability)
P(B) - measure of how often B is observed to occur in general (marginal likelihood)

This algorithm then calculates the probability of a document belonging to a particular class based on the frequency of each word in the document and the frequency of every single word in the training data for each class.

To train the Multinomial Naive Bayes model, we need a labelled dataset that contains a set of documents and their corresponding classes. The algorithm then uses this dataset to calculate the prior probability of each class and the probability of each word occurring in each class. Once the model is fully trained, the model can be used to predict the class of new documents by calculating the probability of the document belonging to each class and choosing the class with the highest probability.

One of the advantages of the Multinomial Naive Bayes algorithm is that it can handle a large number of features (i.e., words) efficiently, making it suitable for text classification tasks with a large vocabulary. It is also relatively simple to implement and can be trained quickly on large datasets. However, it may not perform as well as other more complex algorithms in some cases.

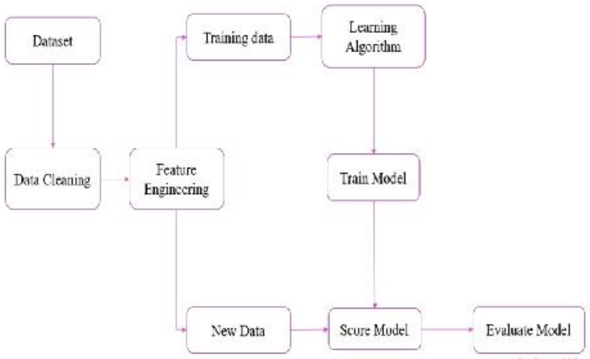

Fig. 1 Flowchart of Multinominal Naïve Bayes

1) Data Analysis

Data analysis refers to the process of inspecting, cleaning, transforming data to extract useful information and insights.

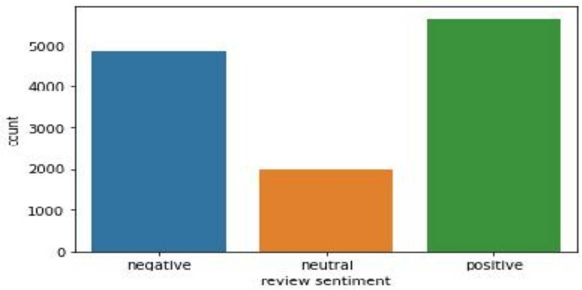

Fig.2 Graph of review distribution

The review sentiment class (negative, neutral, positive) distribution is plotted.

The proportion Target Class is checked and by checking we get:

Positive: 45%
Neutral: 16%
Negative: 39%

Now we are renaming the target variable values. Instead of using 0, 1 and 2 as values of target variable, we can use more appropriate values like "Negative", "Neutral" and "Positive".

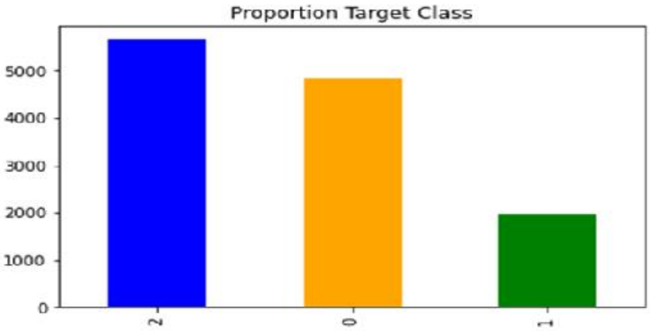

Fig.3 Graph of Proportion Target Class

2) Pre-process:

The main goal of pre-processing is to prepare the input data in a way that is appropriate for the algorithm and can improve the accuracy and efficiency of the model. most common words in target class variable still have untidy data. Therefore, before modelling the data we use must be pre-processed.

[i] *Expanding Contractions*

Contractions is kind of word like “we’d’ve, can’t” etc. By expanding the contractions “we’d’ve” will become “we would have”, “can’t” will become “cannot”, and so on.

[ii] *Uppercase to Lowercase*

All uppercase characters will be changed to lowercase characters. For example: “Good, EXCELLENT” will be changes to “good, excellent”.

[iii] *Removing special characters*

The special characters that are present in the text like @#$! must be removed.

[iv] *Replace elongated words*

All the elongated words will be replaced with the appropriate words. For example: “soooo, loooong” will be replaced with “so, long”.

[v] *Removing stop words*

Removing the stop words have significant effect in building this model. Stop words are words that are commonly used in a language but do not contribute much to the meaning of the complete sentence or document. For example: “The mobile phone is looking good”. After removing stop words the sentence will look like “mobile phone looking good”.

[vi] *Stemming*

Stemming aims to reduce words to their root or stem form. For example: The word “working”, “worked” will be stemmed into the root word “work”. Stemming is simply transforming target words into source form.

[vii]*Drop numbers*

The numbers that are present in the text are removed, since numbers doesn't give much importance to get the main words.

BERT Algorithm

BERT stands for Bidirectional Encoder Representations from Transformers. BERT-based language model is successful in applications that requires a deep understanding of the languages and perform sentiment analysis or text classification. It is designed to help computers understand the meaning of ambiguous language in a text document. The more words that are present in the reviews the more ambiguous the word becomes and here the BERT algorithm comes in handy.

This model is proficient in understanding the context and interpreting patterns without understanding the language completely. It can process large amounts of reviews and languages as in service and product review datasets and it works well for task-specific models. This can be applied directly to the data with no further training and still deliver a high-performing model. BERT model has been trained on a large corpus, making it easier for smaller and more defined tasks. The accuracy of this BERT model is outstanding because it is frequently updated.

BERT has one of the complete architectures for performing various natural language tasks. The BERT gives the most efficient and the most flexible representation for sequences. BERT has three parts:

a) *Bidirectional*

Usually, language models only read text language input as either left-to-right or right-to-left but they couldn't read from both directions simultaneously. In contrast, BERT is bidirectional meaning it has access to the whole sentence. Being a bidirectional model is the key reasons that Transformer based models like BERT achieves impressive results in NLP tasks.

b) *Encoder Representation*

The encoder representations refer to the output of the final layer of the BERT model. BERT is a transformer-based model that consists of multiple encoder layers. Each encoder layer has a multi-head attention mechanism and a feed-forward neural network. The multi-head attention mechanism allows the model to attend to different parts of the input sequence. The feed-forward neural network will allow the model to process the attended information.

c) *Transformers*

This model is based on transformers as this deep learning model every output is connected to every input. BERT makes use of the Transformer part as it learns the contextual relations between words in a text that is it learns how important a word is in a sentence by looking at their specific positions in a sentence. BERT processes each masked token in the Encoder, it does not need a Decoder. BERT model only uses the Encoder part and not the decoder part of the transformer.

1) Data Analysis

Firstly, the review scores are counted. Each product is rated by the customers and the ratings for the products are given from 1 to 5.

1 - being the lowest rating given to the product.
5 - being the highest rating given to the product.

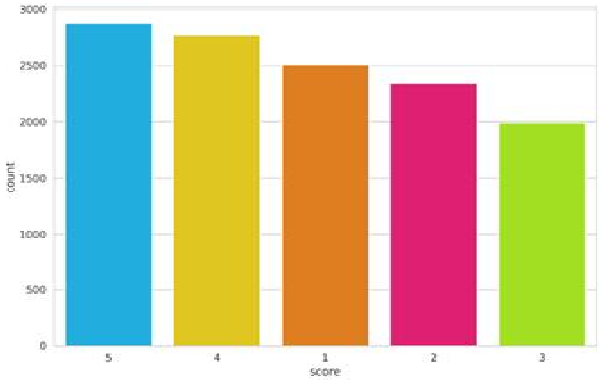

Fig.4 Graph of review scores

The sentiment score is then used to predict the sentiment of the input text (positive, negative, or neutral sentiments). Then the review scores are converted into sentiment classes. The ratings below 2 are classified under negative sentiment classes. The ratings equal to 3 are classified under neutral sentiment classes. The ratings above 3 are classified under positive sentiment classes. With the scores the sentiment classes are classified. These counts of the review sentiments are classified and plotted.

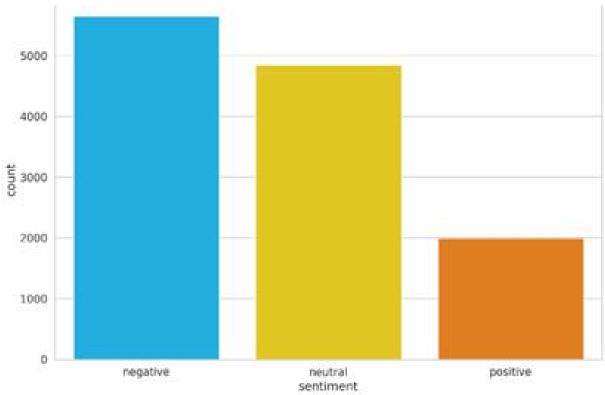

Fig.5 Plotting the review sentiment distribution

2) Tokenization

BERT model takes the token (words) as inputs. The tokenizer plays a critical role in the BERT model, as it determines the vocabulary size, sequence length, and input format of the text data. The tokenizer is responsible for breaking down the input text into a sequence of words (tokens) or subwords, which are then assigned unique numerical values.

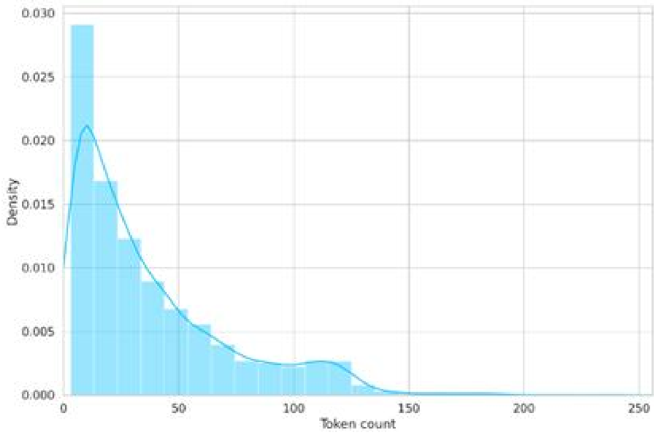

Fig.6 Graph of tokenization

The review length distribution is plotted. Here token count refers to the number of tokens or subwords that are generated by the tokenizer for a given input text. The token count is needed because it will determine the size of the input that will be fed into the BERT model for processing.

Density refers to the distribution of tokens or subwords in a text corpus. Specifically, the density refers to the frequency with which different tokens appear in the corpus, and how evenly or unevenly these tokens are distributed.

IV. IMPLEMENTATION

The training and testing phases are important steps in building and evaluating a model. Training and testing are essential steps in Sentiment Analysis, which help in the process of determining the sentiments of a given piece of text.

The efficiency of the Sentiment Analysis model is typically evaluated using metrics such as accuracy, training history and epoch. These metrics provide an indication of how well the model is able to correctly identify the emotion in the review.

In summary, training and testing are essential steps in sentiment analysis, which involve using labelled data to train a machine learning model to recognize the sentiment of text and it evaluates the model's performance on a separate set of data. These steps are critical for ensuring that the model is accurate and generalizes well to new, unseen data.

A. Multinomial Naïve Bayes

The next step after pre-processing the dataset is to create a numerical feature vector for each document and split the dataset into training and testing data.

The training phase involves using a set of labelled data (i.e., a set of input features and corresponding class labels) to estimate the parameters of the MNB model. This involves computing the prior probabilities of each class and the conditional probabilities of each feature given each class, based on the frequency of the feature occurrences in the training data. This step is usually done using maximum likelihood estimation.

Once the model is trained, the model will be used to predict the class labels of new, raw or unseen data in the testing phase. The testing phase involves using a separate set of labelled data to evaluate the performance of the model by comparing the predicted labels to the actual labels of the test data. The efficiency of the model is typically measured using metrics such as the confusion matrix and accuracy.

Overall, the training and testing phases are critical for building and evaluating the performance of the MNB model, and they help to ensure that the model is accurate, robust, and generalizes well to new, unseen data.

After we get the result from the modelling, the next step is evaluating the model using matrix evaluation. In MNB classification, a confusion matrix is a table that will summarize the performance of the model by comparing the predicted labels to the actual labels of a set of test data. It is a commonly used evaluation metric to assess the accuracy of the model.

The confusion matrix is a square matrix that has the same number of rows and columns as in the number of classes in the dataset. The rows of the matrix represent the true labels of the testing data, while the columns represent the predicted labels of the model.

True Positive (TP): Predicted positive and the actual's positive.
True Negative (TN): Predicted negative and the actual's negative.
False Positive (FP): Predicted positive but the actual's negative.
False Negative (FN): Predicted negative but the actual's positive.

From looking into the confusion matrix, the best evaluation metrics like accuracy, precision and recall can be calculated.

Accuracy: Percentage of how much classes were predicted correctly out of the total.

Precision: Percentage of how much the actual positive classes were predicted correctly out of the total.

Recall: Percentage of how much out of all the predicted positive classes were actually positive.

In MNB, the ROC curve can be constructed by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR)for different values of the threshold. The TPR represents the distribution of positive instances that are correctly identified by the classifier, while the FPR represents the distribution of negative instances that are incorrectly classified as positive.

The ROC curve is a useful tool for evaluating the efficiency of the Multinomial Naive Bayes classifier, as it allows one to visualize the trade-off between the TPR and FPR at different classification thresholds. A good classifier will have a ROC curve that approaches the top-left corner of the plot, which indicates high TPR and low FPR at all threshold values.

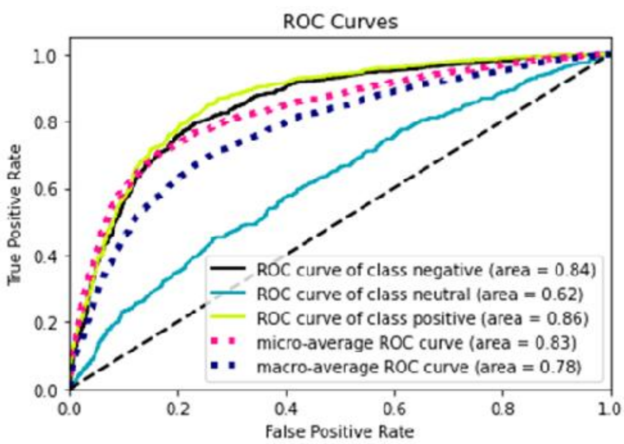

Fig.7 ROC curve

From the Figure 7, it shows that the curve is closer to the top-left corner and it is indicating that the model has a better performance.

B. BERT Model

The training data is an initial set of data used to train the BERT model. The review data set that is cleaned is then pre-processed. The data now should be given polarized value like “positive”,” negative” and “neutral”. Training datasets are fed to BERT algorithm to teach them how to make predictions or analyse the sentiment. Training dataset is also called as a training set, training dataset or learning set.

During training, it is important to monitor the efficiency of the model and adjust hyperparameters as needed to prevent overfitting and improve generalization performance. This can be done by evaluating the performance of the model on a validation set.

For the trained data, training accuracy and validation accuracy will be plotted. The graphs characters will be

1) Training history

The trial and error acquired during training the dataset is the training history. The training history refers to the record of the model's performance during the training process.

2) Accuracy

The performance and accurate predictions of the model. Accuracy is a performance metric that measures how often a model correctly predicts a target variable. In other words, accuracy is the measure of how close the predicted outputs of a model are to the actual outputs.

3) Epoch

During each epoch, the model is fed with the training data in batches, and the weights of the model are updated after each batch. After the model is trained with the entire dataset once, one epoch will be completed. This means the training dataset completes one pass through the model.

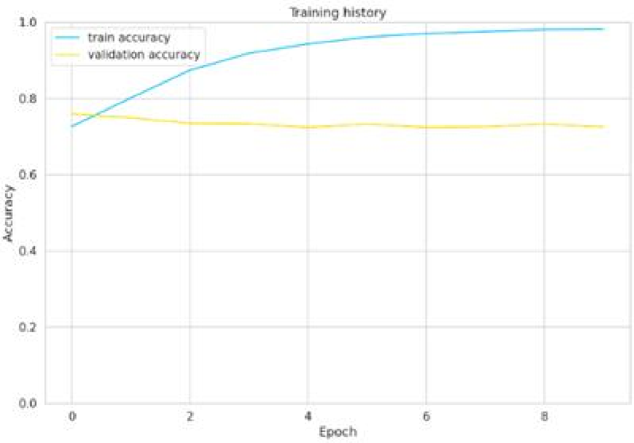

Fig.8 Validation graph

A validation graph is a plot that shows the performance of a model on the validation set during training. The x-axis of the graph typically represents the number of training iterations or epochs, while the y-axis shows the performance metric that is accuracy.

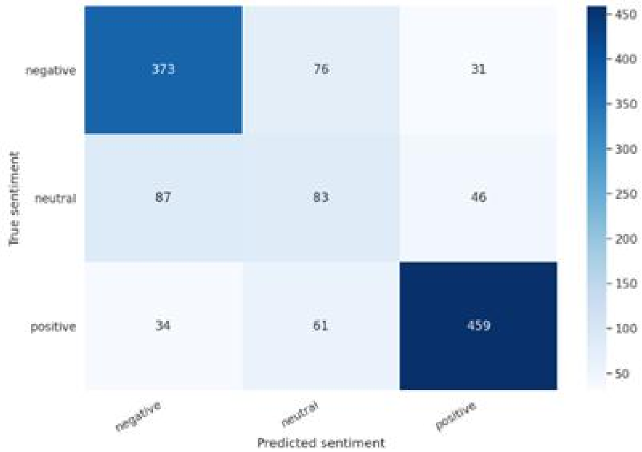

Fig.9 Confusion matrix

V. CONCLUSIONS

This project “Aspect Based Sentiment Analysis of Customer Reviews Using Deep Learning Algorithm” will efficiently perform Sentient Analysis with BERT model. BERT is a contextualized language model that can capture the meaning and context of tokens or words in a review text. This is important for the process of Sentiment Analysis because the emotion of a sentence can be influenced by the context in which it is used. BERT is a powerful model for Sentiment Analysis because it can capture the context and meaning of the text, which is important for accurately predicting sentiment. This means that it can outperform traditional methods for sentiment analysis and can provide more accurate predictions. BERT is also one of the advanced text analysing models.

Overall, BERT provides a powerful tool for Aspect-based Sentiment Analysis, allowing for more accurate analysis of sentiment towards different aspects and entities in the text.

REFERENCES

[1] Kaushal Kumar and B.P. Pande, “Analysis of Customers’ Reviews using Soft Computing Classification Algorithms: A Case Study of Amazon”, Advanced Soft Computing Techniques in Data Science, pp. 331-352. November 2021.

[2] Dr. Shailendra Narayan Singh and Twinkle Sarraf, “Sentiment Analysis on User Reviews of a product with Random Forests Algorithm”, International Conference on Cloud Computing, Data Science and Engineering, 2020.

[3] Alhassan Mabrouk, Rebeca P, Diaz Redondo “Deep Learning Based Sentiment Classification: A Comparative Survey”, IEEE Access, Vol 8, 2020.

[4] B.Seetharamalu, B.Naresh Kuar Reddy and K.Bramha Naidu, “Deep Learning for Sentiment Based on Customer Reviews”, IEEE, July 3, 2020.

[5] B Saritha1, Janmenjoy Nayak, "Aspect Based Sentiment Analysis with Support Vector and Naïve Bayes classifiers", The International Journal of Analytical and Experimental Modal Analysis, Vol XI, August 2019.
[6] Márcio Guia1, Rodrigo Rocha Silva, "Adaptive Comparison of Naïve Bayes, Support Vector Machine, Decision Trees and Random Forest on Sentiment Analysis", International Conference on Knowledge Discovery and Information Retrieval, 2019.
[7] Brij Gupta, "Deep Recurrent Neural Network vs Support Vector Machine for Aspect Based Sentiment Analysis", Journal of Computational Science, pp. 386-393, July 2018.
[8] Adnan, Sohail, "Aspect-Based Sentiment Analysis based on CNN and GA using Hybridized Approach", Institute of Electrical and Electronics Engineers IEEE, 2017.
[9] Min-Yuh Day, Yue Da Lin, "Deep Learning for Sentiment Analysis with Customer Review", IEEE, 2017.