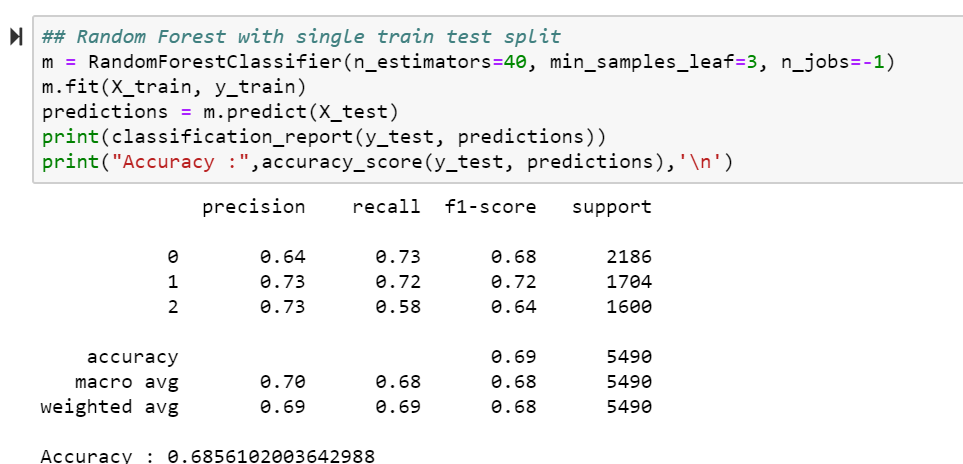
# **Predicting the sentiment of the tweet – either Positive, Negative or Neutral**

**Models Used**

Feature extraction was done using tf-idf vectorizer.

* **Random Forest**

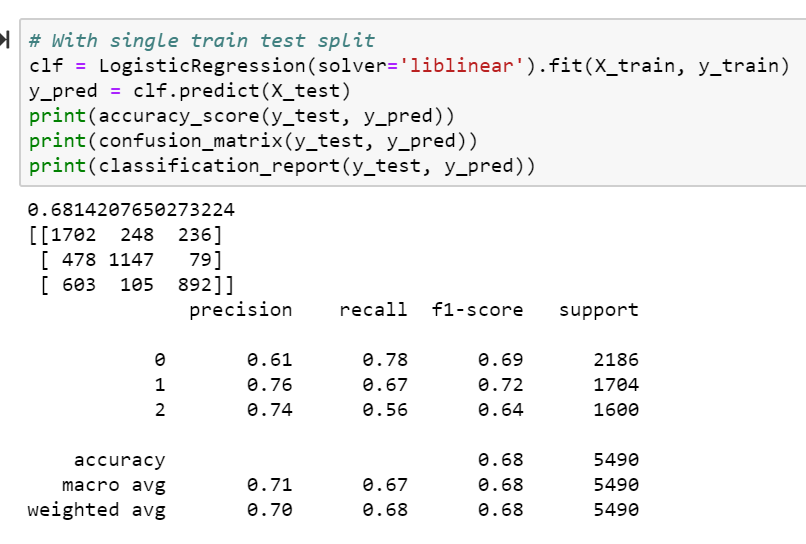


An accuracy of **68.5%** was achieved using random forest with single train test split, n\_estimator value as 40, n\_estimator denotes the number of trees in the forest. A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset. Min\_sample\_leaf (int or float) parameter denotes the minimum number of samples required to be at a leaf node. This may have effect of smoothing the model. N\_jobs denotes the number of jobs to run in parallel. Fit, predict, decision\_path and apply are all parallelized over the trees.

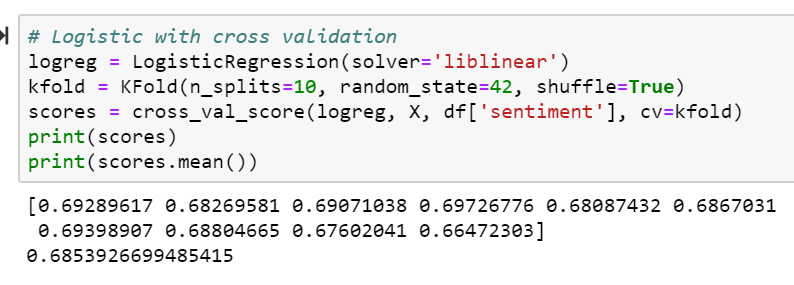
Precision here is found to be 70%, recall is 68% and f1-score is 68% which are pretty close, so this model can be considered to predict labels on validation dataset.

Cross-validation was also performed here but it didn’t improve the accuracy.

* **Logistic Regression**

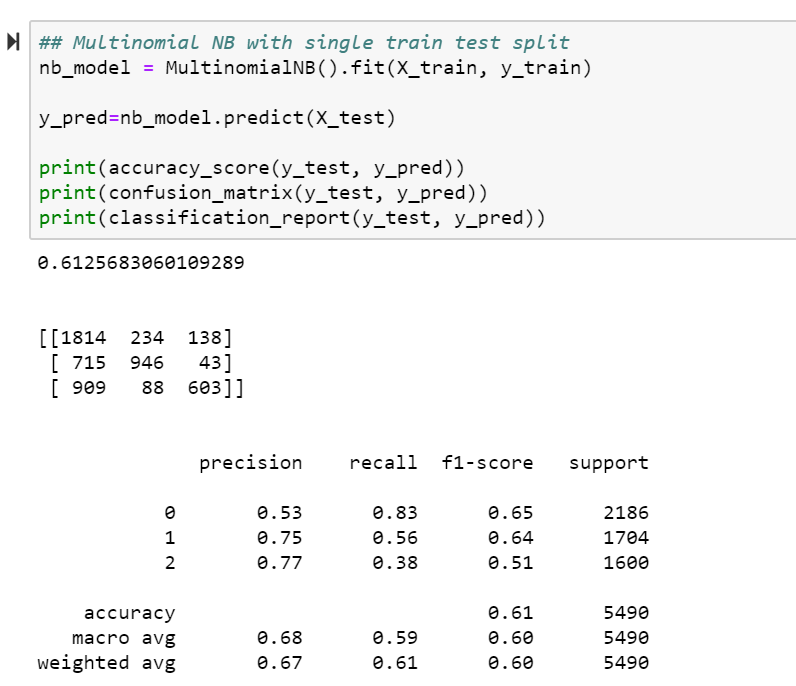


Accuracy achieved here is **68%** with single train test split. Liblinear is used as the solver here, it’s a software package for linear classifier learning. Liblinear can be used for both binary and multiclass classification.



Logistic Regression with K-fold cross-validation gave an accuracy of **68.5%** which is almost similar to the normal logistic regression.

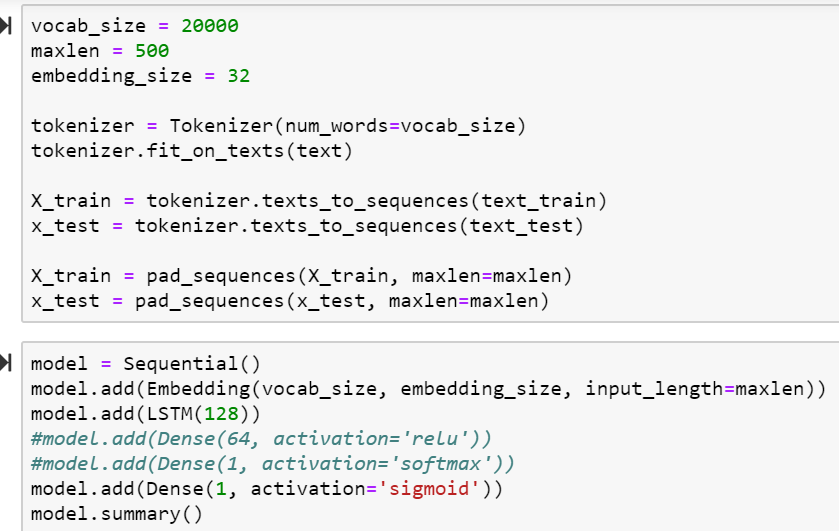
* **Multinomial Naïve Bayes**

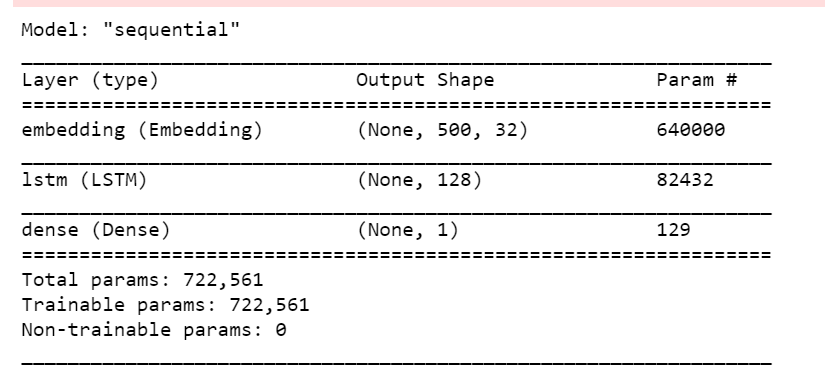


Accuracy achieved here is 61.25%.

The multinomial naïve bayes classifier is suitable for classification with discrete features. The multinomial distribution normally requires integer feature counts but it works also with fraction counts such as tf-idf. Since a Naïve Bayes text classifier is based on the Bayes’ theorem, which helps us compute the conditional probabilities of occurrence of each individual event, encoding those probabilities is extremely useful. This algorithm is mostly used in text classification and with problems having multiple classes.

* **LSTM Model**





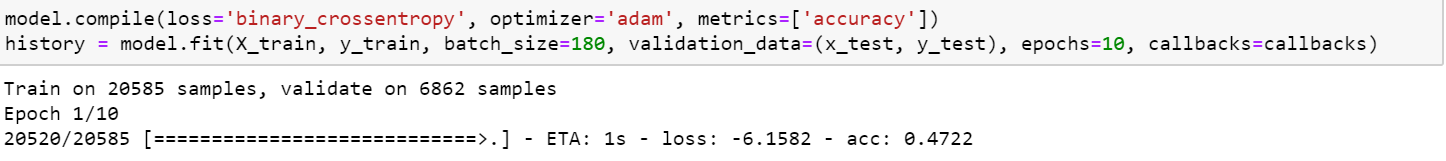
The word embedding layer expects input sequences to be comprised of integers. We can map each word in our vocabulary to a unique integer and encode our input sequences.

To do this encoding Tokenizer class is used. First, the Tokenizer must be trained on the trained dataset, which means it finds all of the unique words in the data and assigns each a unique integer.

We can then fit Tokenizer to encode all of the training sequences, converting each sequence from a list of words to a list of integers.

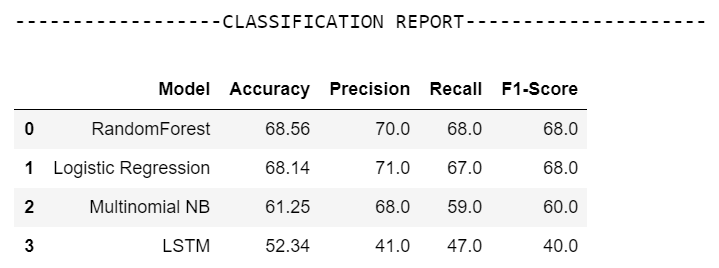
The learned embedding needs to know the size of the vocabulary and the length of input sequences.

One LSTM hidden layer with 128 memory cells is used. More memory cells and a deeper network may achieve better results but here it didn’t. Sigmoid Activation function is used here.

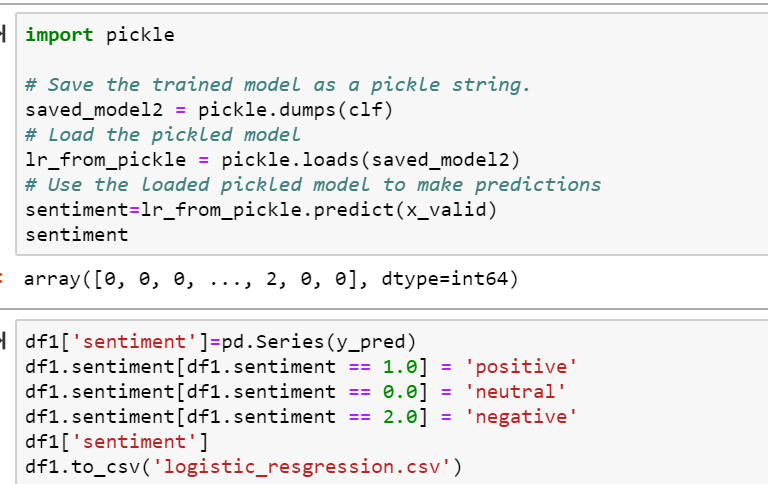


The model is compiled specifying the binary cross entropy loss needed to fit the model. The efficient Adam implementation to mini-batch gradient descent is used and accuracy is evaluated of the model. Accuracy achieved here is **52.3%** only.

**Report**

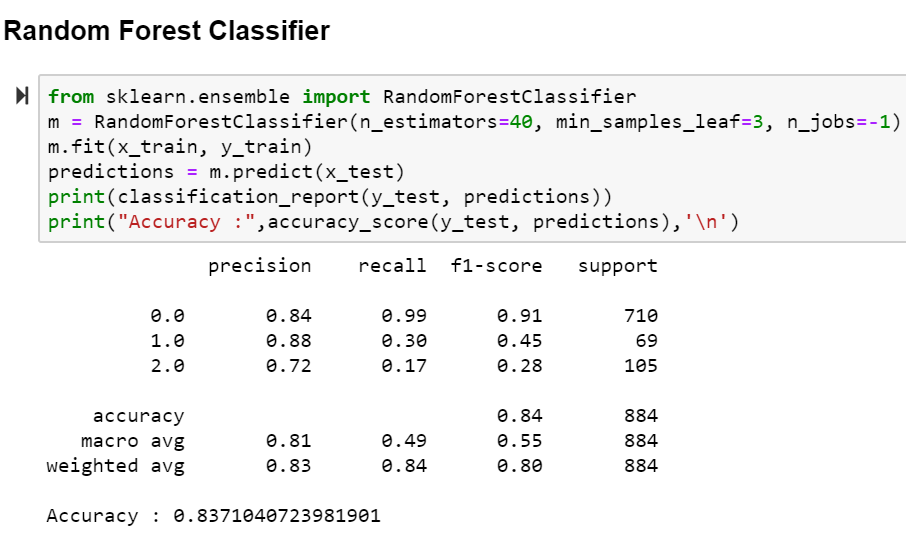
****

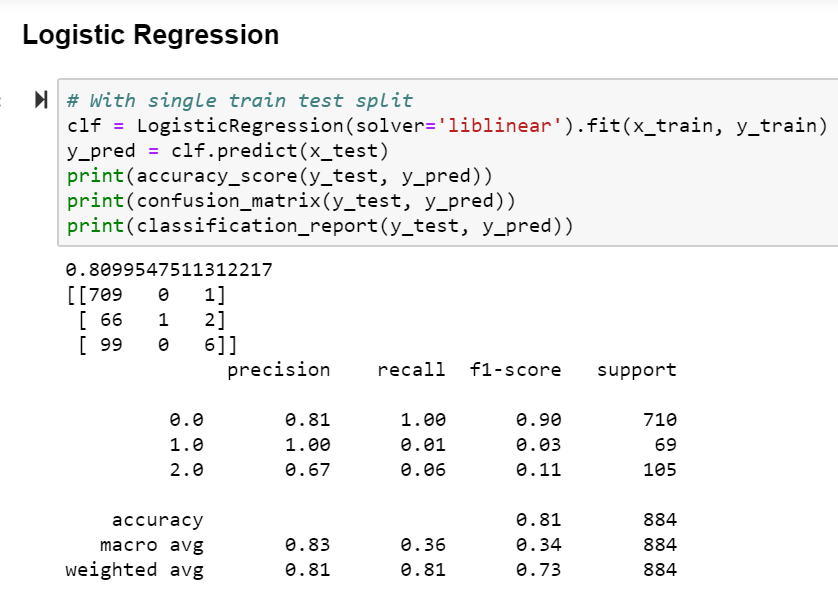
**Predicting Validation Labels**

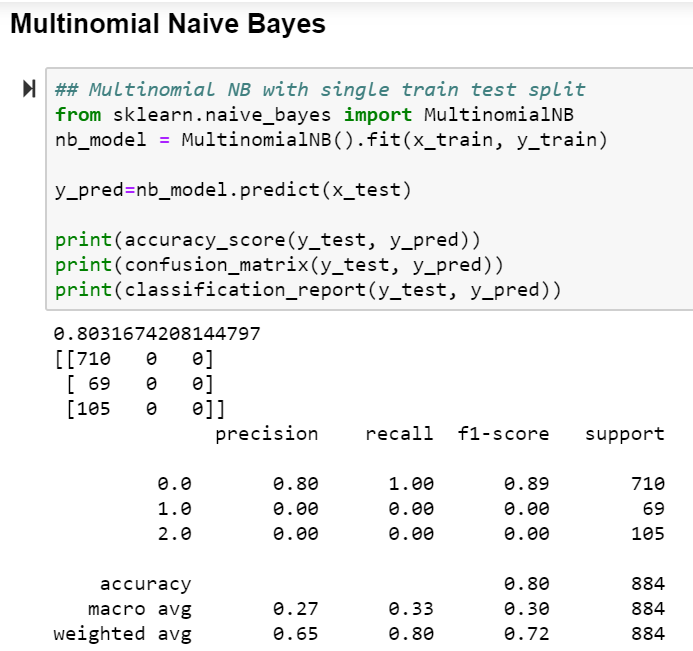


Pickle is used here to save the logistic regression model and load it to predict sentiment labels on validation dataset.

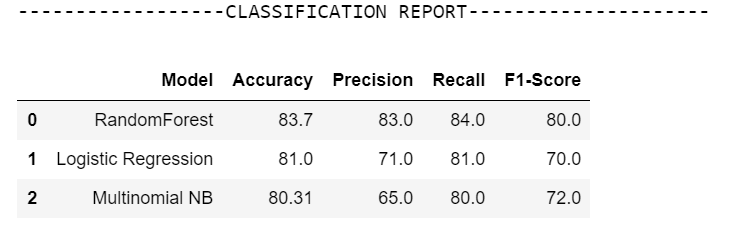
**Modelling on Validation Dataset**







**Report**

****

**Challenges Faced**

* Bag of Words technique for feature extraction was not found to perform well here, hence tf-idf vectorizer was chosen
* Though LSTM model is known to be one of the best algorithms to process text data, here it didn’t perform up to the mark.
* LSTM model with greater batch size was found to consume more memory and execution time, so was not feasible here.

**Further Improvements**

* One of the challenging tasks, Negation handling must be taken care of, because in most the cases that is predicted as a neutral or positive label.
* Similarity based approaches can be applied such as lexicon-based similarity, cosine similarity.
* Correlation based feature extraction methods can be tried out.

**State of the art Sentiment Analysis**

**Introduction**

Sentiment Analysis combines the application of NLP, computational linguistics and Text analysis in its own way. Advanced Sentiment Analysis is the one that goes ‘beyond polarity’ sentiment classification, and it looks for instance, at emotional states such as ‘angry’, ‘sad’ or ‘happy’.

Current state-of-the-art methods rely on features extracted in an unsupervised manner, mainly through one of the existing pre-trained word embedding approaches. These approaches represented words as some function of their contexts, enabling machine learning algorithms to generalize over tokens that have similar representations.

Some of the sub-problems that still are the object of further research attention by the NLP community are: 1) Coreference resolution 2) Negation handling 3) Word sense disambiguation 4) Meaning extraction 5) Optimised parsing

**Traditional Approaches**

The first important step to begin with while handling text data is pre-processing. After the tokenization of a sentence, all word types having an overall occurrence frequency of three or less in the corpus is removed. Because the word types having small occurrence frequency convey more noise than pertinent information. This pruning heuristic also allows to eliminate specific slags that could be hard to classify without any additional lexical vocabulary. To represent a text, bag-of-words assumption is adopted as one of the techniques in which each word is stemmed according to Porter’s method. Such a sequence of words corresponds to one unigram model. POS tagging information is not used in order to reduce the language specific pre-processing of the corpus. The second step is to determine the terms belonging clearly to one category. Z scores calculation was considered for this.

To take into account local proximity, extract the neighbours of each confident feature (two terms before and two terms after). The bigram indexing scheme does not always capture these expressions since a lot of times they may contain modifiers or other words in-between. Sum of the confident features can be checked and classify the sentence accordingly.

To represent textual unit, different strategies are used. First, the unigram model is based on a sequence of stems obtained with the Porter’s stemmer. Second, bigram indexing scheme is used. As a third approach, it is noticed that some of the prepositions combined with the previous term in the sentence can change its meaning, and sometimes even its polarity. Therefore, together with the experiments with unigram and bigram representations, a new indexing scheme called WiseTokenizer is implemented. In this new procedure, the terms in the sentence are indexed separately, except terms that precede the prepositions that could change the meaning of the verb.

In the proposed procedure, the extraction and weighting of confident features are based on Z score model, able to determine term specificity according to two or more categories. Based on the information gain measure, neighbourhood of confident terms are suggested to be considered. The model that was able to achieve comparable results here was classical methods such as SVM and Naïve Bayes.

**Machine learning approach**

In sentiment analysis mostly the supervised learning type is employed to classify the content. It mainly involves 4 main steps- collecting data, preparing data for processing, training and testing data, then finally the classification step. The sentiment classification model is built based on the training data which is then used later with new data inputs. The tweets were pre-processed firstly using tokenization, normalization, and POS tagging. Then, naïve bayes classifier has been used to categorize tweet content as positive or negative by comparing each word with the labelled words in the dictionary.

**Lexicon traditional approach**

The idea behind lexicon-based approach is used to build a dictionary which holds all the positive and negative words and then use it to measure the sentiments of given text based on the appearance of positive and negative words on the input text. The process starts by converting the input text into tokens then scan all tokens, and if the encountered token matches the lexicon in the dictionary, the sentiment score is changed on the type of word. Negation handling has been found to be challenging task in sentiment analysis. Therefore, fuzzy logic is proposed as solution to this.

The main differences between machine learning technique and lexicon-based technique was based on text pre-processing, employed method, dictionary, datasets, and soft-computing approaches.

One of the researches has provided a lexicon-based method to measure and analyse the sentiment of English customer reviews in twitter data. Their method is based on the following rule: the collective polarity of a document or sentence is the polarities sum of the singular words or phrases. They found this approach resulted in a high accuracy sentiment analysis, and they recommended the use of hybrid approaches in future to improve the performance.

**Deep learning approach**

Deep learning has received a lot of attention recently and for a good reason; it achieved classification and analysing results that were not possible before. As deep learning achieved surprising performance on tasks such as language understanding and image analysis, it’s believed it might bring the same level of accuracy and performance in the field of sentiment analysis.

One of the researchers applied their deep learning model at the message using Semeval2015 dataset to analyse twitter morale. They proposed a solution that combines, unsupervised learning of text and learning about poorly supervised data. Their system gave high results across all other test sets.

Another researcher, reported three main findings after performing an experiment. First, that deep learning models are not always better than traditional machine learning methods. Second, the classification accuracies in the real-world setting are much lower than in the experimental setting, which means the current reported performance of twitter sentiment analysis in previous studies may be over-claimed. Third, they reported that support vector machine models have rather good fitness to NLP tasks, especially when the size of the dataset is limited.

In one of the experiments, tweets were labelled into three categories: positive, negative and neutral based on the Euclidian distance and the cosmopolitan similarity. Correlation based feature selection was used as well to determine the best classification features. KNN and similarity-based cosine outperformed all other models.

Another system proposed from a different experiment was a hybrid design of both convolutional neural networks and recurrent neural networks to implement the deep learning model. They performed the following main steps to create the model: pre-processing, word embedding and then feeding input text into a deep neural network architecture. They stated that a hybrid architecture of both convolutional neural networks and recurrent neural networks could achieve an optimal accuracy result.

**Conclusion**

The traditional sentiment analysis approach uses a bag-of-words approach which has failed to take into consideration the order of the words and hence leads to a less accurate model. This approach also often fails to classify negative sentences, i.e., sentences with negation and often give it a positive label. On the other hand, deep learning fixes these issues by generating new feature representations, instead of using term frequency-inverse document frequency vector, the model creates its representations of the word. Given that word2vec identify the semantic context of words in a given dataset as numeric vectors, which facilitates find words that have similar semantic meanings. However, it should be noted that the practicality of the neural networks of deep learning is having both large-scale data for training and super-fast computers for processing.

Deep learning is still in early stages for sentiment analysis research field. Even though in most of the cases, the sentiment analysis results for the machine learning based approach outperform deep learning, it’s still believed that deep learning approach needs more investigation to prove their effectiveness, capability and limitations. Taking in consideration that deep learning automatically finds out the best text features needed for sentiment classification, but this is not the case in machine learning-based approach.