Subject categorization based on questions in Major India Examinations using tenser flow & deep learning

Submitted in partial fulfilment of the requirements for the degree of

Bachelor of Technology

In

Computer Science and Engineering

By

Prabhakar kumar(18BCE0194)

Parth Patel(18BCE0806)

Ajit Singh (19BCE2096)

Ayush Tambi (19BCE2142)

Under the Guidance of Prof. RAJESHKANNAN R

CSE4022 Natural Language Processing

VIT

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

DECLARATION BY THE CANDIDATE

We hereby declare that the project report entitled **“Subject categorization based on questions in Major India Examinations using tenser flow & deep learning”** submitted by us to Vellore Institute of Technology, Vellore in partial fulfilment of the requirement for the award of the degree of **B. Tech in CSE** is a record of J- component of project work carried out by us under the guidance of **Prof. RAJESHKANNAN R.**

We further declare that the work reported in this project has not been submitted and will not be submitted, either in part or in full, for the award of any other degreeor diploma in this institute or any other institute or university.

Place: Vellore Institute of Technology, Vellore.
Date: 25-11-2021

1.ABSTRACT:

Due to the increased availability of hardware and software tools for generating digital data (e.g., personal computers, digital cameras, word processors) and digitizing data that had been created in nondigital form over the last 15 years, the production of digital documents has surged (e.g., scanners, OCR software). This phenomenon has had a significant impact on "new" digital media including picture, video, music, and so on. Natural language text, on the other hand, has been the medium most responsible for this boom, at least from a quantitative standpoint, due to its immediacy and the prevalence of word-processing and text authoring tools.

As a result, there is a greater demand for hardware and software solutions for storing, organizing, and retrieving the massive amounts of digital text generated, with an eye toward its future use.

Natural Language Processing (NLP) has been magically changed by the deep neural network (DNN). The two variants of neural networks, the convolutional neural network (CNN) and the recurrent neural network (RNN) can effectively cope with the different tasks of NLP. CNN relies on feature extraction using top-level n-grams. The RNN can be used effectively to model sequential information. Choosing a neural technique for various NLP tasks is always a challenge. NLP allows computers to understand natural language like humans do. Regardless of whether the language is spoken or written, natural language processing uses machine learning to capture, process, and derive information from the real world so that a computer can understand it. There are mainly two approaches to performing NLP operations: the rule-based approach and the machine learning-based approach. In a rule-based approach, the computer uses human-defined semantic rules. Whereas in a machine learning-based approach, the computer uses machine learning and statics to manage the language. In this project we use a machine learning approach, namely neural networks, to perform NLP. And it also enables developers to create machine learning programs using a variety of tools, libraries, and community resources. Artificial Intelligence is reshaping text classification techniques to better acquire knowledge. However, in spite of the growth and spread of AI in all fields of research, its role with respect to text mining is not well understood yet.

2.INTRODUCTION:

We are a country of more than 1,300 million people. Even if we look at a medium-sized company, it will have a considerable student base. Many students struggle with different problems during their studies. An education technology company would have too many student-teacher doubts. This can lead to problem solving and doubts. Any delay in the questioning process is a critical flaw in online educational philosophy and would affect a student's dynamism. To remedy this situation, we propose an NLP solution. This project would identify the category of doubt related to your subject so that it can be referred to an appropriate subject teacher. This will reduce the delay in clearing up any questions and help the educational technology company serve many students efficiently at the same time. The Project basically focusses on two methods tenser flow and deep learning. Deep learning falls under the category of Artificial Intelligence where it can act or think like a human. Normally, the system itself will be set with hundreds or maybe thousands of input data in order to make the 'training' session to be more efficient and faster. It starts by giving some sort of 'training' with all the input data. TensorFlow is an end-to-end open-source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications**.** Text classification is a construction problem of models which can classify new documents into pre-defined classes. Currently, it is a sophisticated process involving not only the training of models, but also numerous additional procedures, e.g., data pre-processing, transformation, and dimensionality reduction. Text classification remains a prominent research topic, utilising various techniques and their combinations in complex systems. Furthermore, researchers are either developing new classification systems or improving the existing ones, including their elements to yield better results, i.e., a higher computational efficiency.

2.PROBLEM STATEMENT:

One of the widely used natural language processing task in different business problems is "Text Classification". The goal of text classification is to automatically classify the text documents into one or more defined categories. Some examples of text classification are:

- physics,
- chemistry,
- biology and
- and math subject and more.

So, we can help to coaching and institutes in doubt solving. Even a medium-sized business will have a significant student population. During their studies, many students face a variety of issues. There would be too many student-teacher doubts in an education technology company. This can lead to confusion and problem-solving. Any lag in the questioning process is a major problem in the online educational philosophy, and it will impair the dynamism of the student. We suggest an NLP solution to address this issue. This assignment would identify the type of doubt you have about your subject so that it can be forwarded to a subject teacher who can help you. This will shorten the time it takes to answer any inquiries and allow the educational technology firm to serve a large number of students at once.

4.Literature Review:

1.Title: Text categorization: past and present
Year: [2020]
Structure/Organization – Springer
Scope - The scope of this paper is to compare the conventional models with fizzy logic and deep learning models on various languages and compare the accuracy with each other,
Importance – The authors in this paper compare various models of text categorisation and also tried to assemble the existing work into three basic fields: conventional methods, fuzzy logic-based method, deep learning method. They have also compared how these algorithms work on various languages known to human. They have first used the N-gram approach and compared languages like English using NB and SVM methods, Chinese and Japanese using VSM, Arabic, Urdu and Persian and calculated accuracy of categorisation. Next, they have discussed the Indian languages and all the languages were treated with bag-of-words and N-gram approaches. Next, they move on to the fuzzy logic category the authors state how various work has been done for non-Indian languages however not much has been done for Indian languages The survey gives an overview of several algorithms used by researchers in the classification phase, including fuzzy similarity-based approaches, fuzzy clustering-based methods, defuzzification, and fuzzy incremental models. The final category is the deep learning method where it is explained that how it is easier to categorize unstructured data using machine learning models to develop a robust text categorisation system. However, there are some concerns while working with deep learning techniques such that it is likely to be performed better with a huge volume of data. Also, it is computationally expensive since the training of a deep learning-based system demands high-cost hardware resources.
Chronology – The results show how different methods right from classical conventional methods to deep learning and nature inspired methods of text categorisation have evolved and how the accuracy has improved for various languages be it Indian or non-Indian
Citation - Text categorization: past and present
Ankita Dhar1 · Himadri Mukherjee1 · Niladri Sekhar Dash2 · Kaushik Roy
Conclusion: Using various machine learning techniques, several text classification methods/technologies have been created. However, text categorization encounters difficulties when dealing with the semantic relationships between terms in a text document, making system development difficult. Developing text classification systems

is difficult because obtaining semantic associations from unstructured textual data is difficult. Semantic text categorization methods rely on a number of interconnected factors, and while they have some advantages over others, they also have some drawbacks.

2.Title: Supervised and Semi-Supervised Text Categorization using LSTM for Region Embeddings
Year: [2016]
Structure/Organization – RJ Research Consulting, NY and Big Data Lab, Baidu Inc, Beijing
Scope – The scope of this project is to combine LSTM and CNN models to create a framework of region embedding and pooling and also show how this model works better than individual LSTM and CNN models
Importance – The authors of the paper combine LSTM and CNN models to create a more optimized model. The author has also compared supervised and semi-supervised LSTM models and used one-hot LSTM to investigate region embeddings within the broad framework of 'region embedding + pooling' for text classification. The region embedding of one hot LSTM was comparable to or better than that of the state-of-the-art one-hot CNN, demonstrating its efficacy. They also discovered that models using any of these two forms of region embedding outperformed other techniques, including earlier LSTM models. Combining the two forms of region embedding trained on unlabelled data yielded the greatest results, implying that their strengths are complimentary. As a consequence, we reported significant improvements on benchmark datasets compared to prior best findings.
Chronology – The authors compare first supervised learning of LSTM models and CNN models and later combined the both, they then proceeded to semi supervised models and did the same and compared the results
Citation - http://proceedings.mlr.press/v48/johnson16.pdf
Supervised and Semi-Supervised Text Categorization using LSTM for Region Embeddings – Rie Johnson and Tong Zhang
Conclusion: The findings suggest the following at a high level. First, text area embeddings, which may express higher-level notions, are more useful on this job than single-word embeddings. Second, by working directly with one-hot vectors on labelled or unlabelled data, meaningful region embeddings may be learnt. Finally, a possible future option may be to look for novel region embedding methods with complementing benefits inside this framework**.**

3.Title: Text classification based on hybrid CNN-LSTM hybrid model
Year: [2012]
Organization: IEEE
Author: Xiangyang She; Di Zhang
Scope: convolutional neural network CNN can only abstract local information, cannot well express context information, long short-term memory network LSTM can abstract context dependencies, and the classification result is good, but the training time is extended, a text classification algorithm based on hybrid CNN-LSTM hybrid model is proposed.

Importance: The algorithm uses the Skip-Gram model and the CBOW (continuous bag-of-words) model in word2vec to denote words as vector, using CNN to abstract local features of text, LSTM keeps historical information, extracts contextual dependencies of text, and uses the feature vector output by CNN as the input of LSTM, using SoftMax classifier for classification. Main goal of wod2vec is converting words in text into vector form, map words to a new space, and its core model is a typical neural network model.

Chronology: Word2vec is a three-layer neural network structure, which includes the Skip-Gram model and the CBOW model, respectively. Both models include an input layer, a projection layer, and an output layer. The Skip-Gram model uses the current word to predict the context, while the CBOW model uses the context to predict the current word. The experimental data comes from the condensed version of the full-net Chinese news data set published by the Sogou Lab on the Internet.

Citation:

X. She and D. Zhang, "Text Classification Based on Hybrid CNN-LSTM Hybrid Model," 2018 11th International Symposium on Computational Intelligence and Design (ISCID), 2018, pp. 185-189, doi: 10.1109/ISCID.2018.10144.

Link: https://ieeexplore.ieee.org/abstract/document/8695507

Conclusion: Compared with traditional machine learning methods, it avoids manual feature extraction and reduces dimensionality, and enhances learning ability and self-mining feature ability. Compared with CNN and LSTM combined with word2vec method, the hybrid model can extract text context dependencies better, improve the precision, optimize the text classification performance, and has certain reliability. Finally, we designed a word vector dimension comparison experiment and a time comparison experiment, both show that the hybrid model proposed in this paper has a higher advantage in the field of text.

4.Title: Support Vector Machines for Text Categorization

Year: [2018]

Organization: IEEE

Author: A. Basu; C. Walters; M. Shepherd

Scope: paper is examining whether automatic classification of news texts can be improved by a prefiltering the vocabulary reducing feature set used in the computations.

Importance: First, author compares artificial neural network and support vector machine algorithms for use as text classifiers of news items. Then identify a reduction in feature set that provides better results. Information retrieval systems have used conventional classification schemes while most clustering algorithms use the vector space model forming clusters of documents. The vector space model uses a sparse matrix of keyword occurrences which requires rebuilding for each new set of documents. The machine learning process is commenced by an analysis of sample documents to determine the minimal feature set that generates the expected categorization outcomes. This training phase may be supervised or unsupervised.

Chronology: paper assess an artificial neural net algorithm with a support vector machine algorithm for use as text classifiers of news items. It also identifies a reduction in feature set that can be in both algorithms and test if this reduction impacts the performance. Classification performance is calculated using both recall and precision. In this case recall is the percentage of the correct documents that are allocated to a category by the algorithm. Precision is the percentage of documents assigned to a category that belong to that category. Text categorization is a sequence of opposing results and so both micro and macro averaging can be used to create an overall performance across the set of categories used.

Citation:

A. Basu, C. Walters and M. Shepherd, "Support vector machines for text categorization," 36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the, 2003, pp. 7 pp.-, doi: 10.1109/HICSS.2003.1174243.

link: https://ieeexplore.ieee.org/abstract/document/1174243

Conclusion: In the overall difference of SVM and ANN algorithms for this data set the results over all circumstances for both recall and precision indicate there has been a significant difference in the performance of the SVM algorithm over the ANN algorithm and of the reduced feature set over the larger feature set. Paper concludes that the SVM algorithm is less complex than ANNs because the parameter α that constructs the hyperplane is very small.

2.Title: Text Categorization Using Weight Adjusted k-Nearest Neighbor Classification

Organization: springer link

Author: Eui-Hong (Sam) Han, George Karypis, Vipin Kumar

Scope: Paper presents a Weight Adjusted k-Nearest Neighbor (WAKNN) classification that learns feature weights based on a greedy hill climbing technique.

Importance: It presents two performance optimizations of WAKNN that increase the computational performance by a few orders of magnitude, but do not compromise on the classification quality. paper experimentally assessed WAKNN on fifty-two document data sets from a variety of domains and judged its performance against several classification algorithms, such as C4.5, RIPPER, Naive-Bayesian, PEBLS and VSM.

Chronology: Experimental outcomes on these data sets confirm that WAKNN reliably outperforms other current classification algorithms. k-nearest neighbor (k-NN) classification is an instance-based learning algorithm that has shown to be amazingly successful for a variety of problem domains in which underlying densities are not known. This classification paradigm performs well in data sets with multi-modality. Main disadvantage of this algorithm is that it uses all the features while computing similarities between a test document and training documents. In multiple text data sets, small number of features may be useful in categorizing documents and using all the features might affect your performance. One approach to overcome this problem is to learn weights for unique features.

Citation:

Han, Eui-Hong Sam, George Karypis, and Vipin Kumar. "Text categorization using weight adjusted k-nearest neighbor classification." Pacific-asia conference on knowledge discovery and data mining. Springer, Berlin, Heidelberg, 2001.

Link: https://link.springer.com/chapter/10.1007/3-540-45357-1_9

Conclusion: Paper introduces Weight Adjusted k-Nearest Neighbor classification that learns weights for words. WAKNN finds the optimal weight vector using an optimization function that is based on leave-one-out cross validation and a greedy hill climbing technique. WAKNN has better classification results than many other classifiers, but it has a high computational cost. One of the major challenges of the weight adjustment algorithm is how to decrease the high computational cost.

6.Title: Text Categorization and Machine Learning Methods: Current State of the Art
[2012]
Structure – G. Narayan Amma Institute of Technology and Science, Hyderabad
Scope – The scope of this paper is to represent the overall process of text categorisation and state how modern and current State of the Art ML methods help in increasing accuracy of text classification or categorisation
Importance – In this paper the authors aim to use current State of the Art methods in text categorization which is an assortment of ML algorithms. The main model focused here is the Threshold Reduction Model, in which many a document can be send to the classifiers at the lower level if the sub tree classifiers are kept at the lower thresholds and the blocking problem is solved using the top-down approach and to differentiate the degree of blocking a blocking factor is set as a kind of new classifier centric measure. It has also tried to generalize specific properties of recent trends in learning techniques
Chronology – The paper majorly focuses on the recent trends and is using and comparing the current State of the Art methodology for text categorization and classification
Citation - https://computerresearch.org/index.php/computer/article/view/555/555
Text Categorization and Machine Learning Methods: Current State of the Art – By Durga Bhavani Dasari& Dr. Venu Gopala Rao. K, G. Narayanamma Institute of Technology and Science, Hyderabad
Conclusion: In the general distinction of SVM and ANN algorithms for these facts set the outcomes over all instances for each remember and precision imply there was a huge distinction withinside the overall performance of the SVM set of rules over the ANN set of rules and of the decreased characteristic set over the bigger characteristic set. Paper concludes that the SVM set of rules is much less complicated than ANNs due to the fact the parameter α that constructs the hyperplane may be very small

7.Title: Deep learning methods for subject text classification of articles
Year:[2018]
Structure/Organisation: IEEE
Scope: This work presents a method of classification of text documents using deep neural network with LSTM (long short-term memory) units. We have tested different approaches to build feature vectors, which represent documents to be classified: we used feature vectors constructed as sequences of words included in the documents, or, alternatively, we first converted words into vector representations using word2vec tool

and used sequences of these vector representations as features of documents. We evaluated feasibility of this approach for the task of subject classification of documents using a collection of Wikipedia articles representing 7 subject categories. Our experiments show that the approach based on an LSTM network with documents represented as sequences of words coded into word2vec vectors outperformed a standard, bag-of-word approach with documents represented as frequency-of-words feature vectors.

Importance: This paper shows how a deep neural network with LSTM (long short-term memory) units can be used to classify text content. We tried two methods for creating feature vectors, which represent documents to be classified: we used feature vectors created from sequences of words found in the documents, or we converted words into vector representations using the word2vec tool and then used sequences of these vector representations as document features. Using a set of Wikipedia articles representing seven subject groups, we examined the practicality of this approach for the task of subject classification of documents. The strategy based on an LSTM network with documents represented as sequences of words coded into word2vec vectors beat a traditional bag-of-words approach with documents represented as frequency-of-words feature vectors, according to our findings.

Chronology: Paper check a synthetic neural internet set of rules with a aid vector device set of rules to be used as textual content classifiers of information items. It additionally identifies a discount in characteristic set that may be in each algorithm and check if this discount influences the overall performance. Classification overall performance is calculated the use of each don't forget and precision. In this situation don't forget is the proportion of an appropriate files which are allotted to a class with the aid of using the set of rules. Precision is the proportion of files assigned to a class that belong to that category. Text categorization is a series of opposing outcomes and so each micro and macro averaging may be used to create a universal overall performance throughout the set of classes used.

Citation:

P. Semberecki and H. Maciejewski, "Deep learning methods for subject text classification of articles," 2017 Federated Conference on Computer Science and Information Systems (FedCSIS), 2017, pp. 357-360, doi: 10.15439/2017F414.

Conclusion: In the general distinction of SVM and ANN algorithms for these facts set the outcomes over all instances for each remember and precision imply there was a huge distinction withinside the overall performance of the SVM set of rules over the ANN set of rules and of the decreased characteristic set over the bigger characteristic set. Paper concludes that the SVM set of rules is much less complicated than ANNs due to the fact the parameter α that constructs the hyperplane may be very small

8.Title: A study on Image Classification based on Deep Learning and TensorFlow Year:[2019]

Structure/Organisation: Research gate

Scope: This research study about image classification by using the deep neural network (DNN) or also known as Deep Learning by using framework TensorFlow. Python is used as a programming language because it comes together with TensorFlow framework. The input data mainly focuses in flowers category which there are five (5)

types of flowers that have been used in this paper. Deep neural network (DNN) has been choosing as the best option for the training process because it produced a high percentage of accuracy. Results are
discussed in terms of the accuracy of the image classification in percentage. Roses get 90.585% and same goes to another type of flowers where the average of the result is up to 90% and above.

Importance: This research investigates image classification using a deep neural network (DNN), also known as Deep Learning, and the TensorFlow framework. Python is a programming language that is used in conjunction with the TensorFlow framework. The input data is mostly focused on the flower category, with five different varieties of flowers being used in this work. Because it delivered a high percentage of accuracy, deep neural networks (DNN) have been chosen as the best alternative for the training procedure. The accuracy of the image classification is provided as a percentage in the results. The average result for roses is 90.585 percent, while for other types of flowers, the average result is up to 90 percent and higher.

Chronology: the result of this paper depends on the objectives that need to achieved. Other than that, certain parameters also played its roles to determine the accuracy of the image classification by using the deep neural network (DNN). The first result of this research was tested by conducting classification for each of the types of flowers. It can be seen all of the five (5) types of flowers showed up to 90% accuracy in terms of implementation of the system of image classification by using DNN. This happened due to the abundantly set of data that was being used in order to train the model and of course DNN worked excellent when there were lots of data. Then, the system was tested with an image of cow.jpg where actually these images were not included during the training model. The images also were not one of the categories of flowers instead it fell under animals. There were some errors after doing the classification. The errors state that 'Not Found Error' which meant that the images cannot be recognized by the systems as it was not trained so that model trained can
recognize it as an animal named as cow

Citation:

W. W. T. Fok et al., "Prediction model for students' future development by deep learning and tensorflow artificial intelligence engine," 2018 4th International Conference on Information Management (ICIM), 2018, pp. 103-106, doi: 10.1109/INFOMAN.2018.8392818.

Conclusion: In conclusion, this research is about image classification by using deep learning via framework TensorFlow. It has three (3) objectives that have achieved throughout this research. The objectives are linked directly with conclusions because it can determine whether all objectives are successfully achieved or not. It can be concluded that all results that have been obtained, showed quite impressive outcomes. The deep neural network (DNN) becomes the main agenda for this research, especially in image classification technology. DNN technique was studied
in more details starting from assembling, training model and to classify images into categories. The roles of epochs in DNN were able to control accuracy and also prevent any problems such as overfitting. Implementation of deep learning by using framework TensorFlow also gave good results as it is able to simulate, train and classified with up to 90% percent of accuracy towards five (5) different types of flowers that have become

a trained model. Lastly, Python have been used as the
programming language throughout this research since it comes together with framework TensorFlow which leads to designing of the system involved Python from start until ends.

9.Title: Prediction model for students' future development by deep learning and tensor flow artificial intelligence engine
Year:[2019]
Structure/Organisation: IEEE
Scope: Classification and prediction of students' performance in examination are the typical challenges for educators. Various traditional data mining methods such as decision tree and association rules were used to perform classification. In recent years, the rapid development of artificial intelligence and deep learning algorithm provided another approach for intelligent classification and result prediction. In this paper, research on how to use TensorFlow artificial intelligence engine for classifying students' performance and forecasting their future universities degree program is studied. An appropriate and accurate forecast is important for providing prompt advice to student on program and university selection. For a more comprehensive consideration of an all-rounded factors, the deep learning model analysed not only the traditional academic performance including Mathematic, Chinese, English, Physics, Chemistry, Biology and History, but also non-academic performance such as service, Conduct, Sport and Art. A few parameters in TensorFlow engine including the number of intermediate nodes and number of deep learning layers are adjusted and compared. With a data set of two thousand students, 75% of these data are used as the training data and 25% are used as the testing data, the accuracy ranged from 80% to 91%. The optimal configuration of the TensorFlow deep learning model that achieves highest prediction accuracy is determined. This study determined the factors affecting the accuracy of the prediction model.
Importance: Educators face a variety of obstacles, including classifying students and predicting their exam success. To conduct classification, various classic data mining methods such as decision trees and association rules were applied. The rapid development of artificial intelligence and deep learning algorithms in recent years has created another method for intelligent classification and outcome prediction. This work investigates methods to classify students' performance and anticipate their future university degree programme using the TensorFlow artificial intelligence engine. For offering timely guidance to students on programme and university selection, an adequate and accurate forecast is necessary. The deep learning model looked at not just traditional academic achievement, such as math, Chinese, English, physics, chemistry, biology, and history, but also non-academic performance, such as service, conduct, sport, and art, to get a more complete picture of all elements. The number of intermediate nodes is one of the parameters of the TensorFlow engine.
A few TensorFlow engine parameters, such as the number of intermediate nodes and deep learning layers, are tweaked and compared. With a data set of 2,000 students, the accuracy ranged from 80% to 91 percent when 75 percent of the data was used as training data and 25% as testing data. The TensorFlow deep learning model's best setup for achieving the maximum prediction accuracy is identified. The elements affecting

the prediction model's accuracy were investigated in this study.

Chronology:

Experimental effects on those facts units verify that WAKNN reliably outperforms different cutting-edge class algorithms. k-nearest neighbour (k-NN) class is an instance-primarily based totally gaining knowledge of set of rules that has proven to be amazingly a hit for loads of hassle domain names wherein underlying densities aren't known. This class paradigm plays nicely in facts units with multi-modality. Main drawback of this set of rules is that it makes use of all of the functions even as computing similarities among a take a look at record and schooling files. In more than one textual content facts units, small quantity of functions can be beneficial in categorizing files and the use of all of the functions may have an effect on your performance. One method to conquer this hassle is to research weights for particular functions.

Citation:

Abu, Mohd Azlan & Indra, Nurul Hazirah & Abd Rahman, Abdul & Sapiee, Nor & Ahmad, Izanoordina. (2019). A study on Image Classification based on Deep Learning and Tensorflow. 12. 563-569.

Conclusion: Paper introduces Weight Adjusted k-Nearest Neighbour type that learns weights for words. WAKNN unearths the most desirable weight vector the use of an optimization feature this is primarily based totally on leave-one-out move validation and a grasping hill mountaineering technique. WAKNN has higher type effects than many different classifiers, however it has a excessive computational cost. One of the essential demanding situations of the load adjustment set of rules is a way to lower the excessive computational cost.

10.Title: Chinese News Text Classification Based on Machine learning algorithm

Year:[2018]

Structure/organization: IEEE

Scope: The majority of text data originates from the Internet, and it can be crawled by web spiders or other web page capture programmers. There have been a lot of universities or natural science centers so far. Language processing centers with their own Chinese staff Fudan is one of the most well-known Chinese corpora. Sogou laboratory, university news text classification corpus, Chinese Academy of Sciences, news corpus, and News corpus in English, etc. The data should be pretreated after it has been obtained. The first step is to get rid of all the stop words and other nonsense. For rough dimension reduction, use punctuation marks. Face to face deleting the words that have been removed from the massive amount of data The text will have no effect on the categorization outcome. content is clear, data space is reduced, and the a more efficient classification Texts in Chinese and English are distinct. Words and words are separated in an English text.

Importance: The classification of news text is a key technology to process news information. By tagging news texts, each news item can be classified so that readers can choose what they are interested in. News websites can through each reader's reading records recommend the same or relevant type of news to attract more attention.

Chronology: This paper introduces the principle of the text classification and designs system model based on machine learning algorithm. This model trains the classifier according to the existing data, and then classifies the unlabeled news texts by it.

This system is built on the PyCharm platform and is written in Python. Jieba segmentation is a Chinese word segmentation tool. We use the Fudan University news corpus, which is separated into two sets: training and test.

Each set comprises nine categories, with each category including the matching news items. To classify, this system employs the K-neighbour classifier, SVM classifier, and Naive Bayesian classifier from the Sklearn package.

Citation:

F. Miao, P. Zhang, L. Jin and H. Wu, "Chinese News Text Classification Based on Machine Learning Algorithm," 2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2018, pp. 48-51, doi: 10.1109/IHMSC.2018.10117.

Conclusion:

This paper is constructing a Chinese news text-classification system model based on Machine learning. It expounds the K-nearest neighbor, Naïve Bayesian algorithm and SVM algorithm. The results show that support vector classifier with the TF-idf feature attain the highest accuracy**.**

11.Title: Text classification using artificial neural networks

Year: [2018]

Structure/organization: ResearchGate.

Scope: In this study, we created a TFIDF matrix, which will be fed into an artificial neural network, which will learn and then categories texts into preset categories, which will be implemented and detailed in future articles. Because few people have studied neural networks and their working with respect to text categorization in the existing system, we wanted to implement and show how the text could be easily categorized using ANN techniques because they are simple and have many advantages over traditional methods such as Nave Bayes, KNN, SVM, and so on.

Text categorization is a pattern classification technique for text mining. It is also required in the medical text classification, health profession-also spend large amount of time scanning the notes. There are many automatic text categorization techniques like Naïve bayes, SVM, KNN, decision trees etc.

Importance: Preprocessing involves converting the document into plain text and removing unnecessary stop words like prepositions and participles. Once preprocessing is done then the neural network will be trained to categorize the documents into the categories which were prede-fined.

Chronology: After examining all of the text classification approaches and their merits and downsides, we decided to categories the text using ANN because of its benefits and study, which showed that it produced better results than the other strategies stated in the preceding section. One of the key reasons for utilizing the ANN is that it can address issues that cannot be solved linearly or using linear statistical classification techniques. Other characteristics of ANN include its ability to learn in the presence of noise and the fact that it trains rather than programming computational systems to do essential tasks. ANN has the ability to learn and form complex, non-linear associations. It can also deduce unseen connections on unseen data after learning from the inputs and their linkages. Unlike other prediction algorithms, Ann does not impose limits on the input variables**.**

Citation: Prasanna, P. & Rao, Dr. (2018). Text classification using artificial neural networks. International Journal of Engineering and Technology (UAE). 7. 603-606. 10.14419/ijet.v7i1.1.10785.

Conclusion: This is the R-tool implementation of the TFIDF matrix creation of text documents. This is acquired after removing stop words from the documents and after the stemming procedure, which is done after the preprocessing mentioned above. The rows in this matrix represent the document's values, while the columns represent the words.

12.Title: Sentimental Short Sentences Classification by Using CNN Deep Learning Model with Fine Tuned Word2Vec

Year: [2020]

Structure/organization: ScienceDirect

Scope: People daily exchanged their views and opinions in the form of writings, photographs, videos, and speech as the number of social networking web users continues to rise. Because these massive texts come from a variety of sources and people with various mindsets, text categorization remains a critical challenge. The shared opinion is partial, inconsistent, noisy, and expressed in a variety of languages. To overcome such problems, NLP and deep neural network technologies are now commonly used. For effective text classification, Word2Vec word embedding and the Convolutional Neural Network (CNN) approach must be used. The suggested model in this paper perfectly cleaned the data, generated word vectors from a pre-trained Word2Vec model, and used a CNN layer to extract improved features for categorizing short phrases.

Importance: Experiments were conducted on the sentence polarity dataset v1.0 movie reviews, and pre-trained Google News dataset from Word2Vec was used to generate publicly available word vectors. For the goal of validation accuracy, 9595 sentences or snippets were employed as training sets and 1067 snippets or phrases were used as testing sets in this study. TensorFlow, which runs on Python 3.6, was used to analyses the model, while Kera's 2.2.4 was used to train it.

Kera's is a TensorFlow-based neural network API that uses an embedding layer for real-valued vector representation of words.

Chronology: For computing continuous distributed representation of words, the Word2Vec model is utilized. This model is a simple neural network model with one hidden layer that provides an efficient implementation of CBOW and Skip-gram architectures for computing vector representations of words. The Word2Vec model outputs word vectors from a text corpus as input. The word vectors are trained using backpropagation and stochastic gradient descent after the algorithm constructs a vocabulary from the input words.

Citation: Sentimental Short Sentences Classification by Using CNN Deep Learning Model with Fine Tuned Word2Vec

Amit Kumar Sharmaa,∗ , Sandeep Chaurasiaa , Devesh Kumar Srivastavaa

Conclusion: For sentiment and semantic analysis, deep learning approaches are particularly effective and efficient. For tiny sentences in the movie review corpus, the research yields a more accurate outcome for feature extraction using Word2Vec and

CNN approaches. The suggested model has a training sample accuracy of 99.07 percent and a testing sample accuracy of 82.19 percent. The Word2Vec model was trained to generate fine-tuned word vectors for all words in the corpus, which were then fed into the CNN layer. The model has trained better with scattered word vectors as its accuracy has improved while its value has decreased. The Word2Vec word embedding model is used to represent words in a distributed manner. The proposed method will aid in the classification of text small data from social media and YouTube with more accuracy.

LITERATURE SURVEY TABLE

	Author	Title	Methodology	Observation	Result
1	M. Krendzelak and F. Jakab [2015]	Text Categorization with Machine Learning and Hierarchical Structures	Machine learning approach focused on performance and efficiency of text classification with flat and hierarchical structures.	The hierarchical approach appears to bring few benefits, when Naïve Bayes simple classifiers were used to train each hierarchical node independently.	Flat approach seems much less capable of taking advantage of the richer model space. There are still remaining several specific issues related to hierarchical classification.
2	Guibin Chen, Deheng Ye, Zhenchang Xing, Jieshan Chen and Erik Cambria [2017]	Ensemble application of convolutional and recurrent neural networks for multi-label text categorization	ensemble application of convolutional and recurrent neural networks to capture both the global and the local textual semantics and to model high-order label correlations while having a tractable computational complexity.	Existing approaches to multi-label text categorization fall short to extract local semantic information and to model label correlations.	Evaluations reveal that the power of the proposed method is affected by the size of the training dataset. If the data size is too small, the system may suffer from overfitting. However, when trained over a large-scale dataset, the proposed model can achieve the state-of-the-art performance.
3	X. She and D. Zhang [2018]	Text Classification Based on Hybrid CNN-	wod2vec is an algorithm that transforms words in text	Word2vec is a three-layer neural network	The hybrid model proposed in this paper has a higher advantage in the

		LSTM Hybrid Model	into vector form. It uses a typical neural network model to abstract local features of text. LSTM keeps historical information, extracts contextual dependencies of text, and uses the feature vector output by CNN as the input for classification.	structure, which includes the Skip-Gram model and the CBOW model. The experimental data comes from the condensed version of the full-net Chinese news data set published by the Sogou Lab on the Internet.	field of text. It avoids manual feature extraction and reduces dimensionality, and enhances learning ability. Compared with CNN and LSTM combined with word2vec method, the hybrid model can extract text context dependencies better.
4	Ankita Dhar1 · Himadri Mukherjee1 · Niladri Sekhar Dash2 · Kaushik Roy [2020]	Text categorization : past and present	this paper compare various models of text categorisation and also tried to assemble the existing work into three basic fields: conventional methods, fuzzy logic-based method, deep learning method. They have also compared how these algorithms work on various languages known to human. They have first used the N-gram approach and compared languages like	The survey gives an overview of several algorithms used by researchers in the classification phase. They have also compared how these algorithms work on various languages known to human. The authors state how various work has been done for non-Indian languages however not much attention has been paid to Indian languages.	Developing text classification systems is difficult because obtaining semantic associations from unstructured textual data is difficult. Semantic text categorization methods rely on a number of interconnected factors, and while they have advantages over others, they also have some advantages ove r others.

			English using NB and SVM methods, Chinese and Japanese using VSM, Arabic, Urdu and Persian and calculated accuracy of categorisation. Next, they have discussed the Indian languages and all the languages were treated with bag-of-words and N-gram approaches.		
5	Rie Johnson and Tong Zhang [2016]	Supervised and Semi-Supervised Text Categorization using LSTM for Region Embeddings	Researchers have combined LSTM and CNN models to create a more optimized model for text classification. They found that models using any of these two forms of region embedding outperformed other techniques, including those trained on unlabelled data.	It is to combine LSTM and CNN models to create a framework of region embedding and pooling and also show how this model works better than individual LSTM and CNN models	Text area embeddings may express higher-level notions than single-word ones, a study has suggested. By working directly with one-hot vectors on labelled or unlabelled data, meaningful region embeds may be learnt. Future options for novel region embedding methods could be explored within this framework.
6	Durga Bhavani Damari & Dr. Venue	Text Categorization and Machine	In this paper the authors aim to use current State	This paper is to represent the overall process of	The SVM set of rules is much less complicated than ANNs due to the fact

	Gopala Rao. K [2012]	Learning Methods: Current State of the Art	of the Art methods in text categorization. The main model focused here is the Threshold Reduction Model. It has also tried to generalize specific properties of recent trends in learning techniques into text-based classification.	text categorization and state how modern and current State of the Art ML methods help in increasing accuracy of text classification or categorization. The paper majorly focuses on the recent trends and is using and comparing the current methodology for text categorization and classification.	that the parameters that construct the hyperplane may be very small. Paper concludes that the SVM algorithms are superior to ANNs in their precision and rememberability over all instances for each remember and precision.
7	P. Semberecki and H. Maciejewski [2018]	Deep learning methods for subject text classification of articles	This paper shows how a deep neural network can be used to classify text content. The strategy based on an LSTM network beat a traditional bag-of-words approach, according to our findings. Using a set of Wikipedia articles representing seven subject groups, we examined the practicality of this approach.	Paper check is a synthetic neural internet set of rules with a aid vector device. It identifies a discount in characteristic set that may be in each algorithm and check if this discount influences the overall performance. Text categorization is a series of opposing outcomes, so each micro	The SVM set of rules is much less complicated than ANNs due to the fact that the parameters that construct the hyperplane may be very small. Paper concludes that the SVM algorithms are superior to ANNs in their precision and rememberability over all instances for each remember and precision.

				and macro averaging may be used to create a universal overall performance throughout the classes used.	
8	W. W. T. Fok et al	A study on Image Classification based on Deep Learning and TensorFlow	. The system was tested with an image of cow. The images were not one of the categories of flowers instead it fell under animals. This happened due to the abundantly set of data that was being used in order to train the model and of course DNN worked excellent when there were lots of data.	Python is used as a programming language because it comes together with TensorFlow framework. The input data mainly focuses in flowers category which there are five (5) types of flowers that have been used in this paper. Deep neural network (DNN) has been choosing as the best option for the training process.	WAKNN unearths the most desirable weight vector the use of an optimization feature this is primarily based totally on leave-one-out move validation and a grasping hill mountaineering technique. WAKNN has higher type effects than many different classifiers, however it has an excessive computational cost.
9	Abu, Mohd Azlan & Indra, Nurul Hazirah & Abd Rahman, Abdul & Sapiee, Nor & Ahmad,	Prediction model for students' future development by deep learning and tensor flow artificial	k-nearest neighbour (k-NN) class is an instance-primarily based on totally gaining knowledge of set of rules. This class	A study investigates ways to classify students' performance and anticipate their future university degree	WAKNN unearths the most desirable weight vector the use of an optimization feature this is primarily based totally on leave-one-out move validation and a grasping hill mountaineering

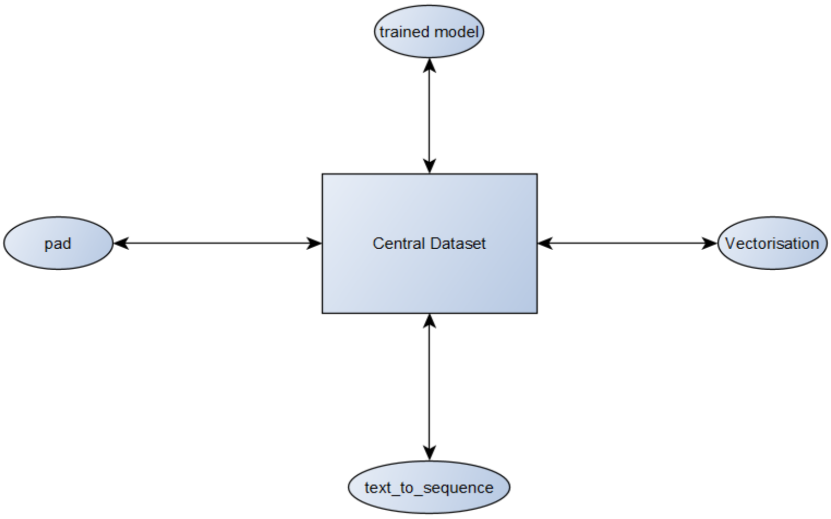
	Izanoordina. [2019]	intelligence engine	paradigm plays nicely in facts units with multi-modality. Experimental results show that WAKNN reliably outperforms different cutting-edge class algorithms	programme using the TensorFlow artificial intelligence engine. With a data set of 2,000 students, the accuracy ranged from 80% to 91% when training data was used as training data.	technique. WAKNN has higher type effects than many different classifiers, however it has an excessive computational cost.
10	F. Miao, P. Zhang, L. Jin and H. Wu [2018]	Chinese News Text Classification Based on Machine learning algorithm	This system is built on the PyCharm platform and is written in Python. We use the Fudan University news corpus, which is separated into two sets: training and test. To classify, this system employs the K-neighbour classifier, SVM classifier and Naive Bayesian classifier.	The majority of text data originates from the Internet, and can be crawled by web spiders. Texts in Chinese and English are distinct. The first step is to get rid of all the stop words and other nonsense. Face to face deleting the words that have been removed from the massive amount of data.	This paper is constructing a Chinese news text-classification system model based on Machine learning. It expounds the K-nearest neighbour, Naïve Bayesian algorithm and SVM algorithm. The results show that support vector classifier with the TF-idf feature attain the highest accuracy.
11	Prasanna, P. & Rao, Dr. [2018]	Text classification using artificial neural networks	This is the R-tool implementation of the TFIDF matrix creation of text documents. This is	In this study, we created a TFIDF matrix, that will be fed into an artificial neural	Unlike other prediction algorithms, Ann does not impose limits on the input variables. Ann can also deduce unseen connections on

			acquired after removing stop words from the documents and after the stemming procedure, which is done after the preprocessing mentioned above. The rows in this matrix represent the document's values, while the columns represent the words.	network, which will learn and then categories texts. It is also required in the medical text classification, health profession-also spend large amount of time scanning the notes.	unseen data after learning from the inputs and their linkages.
12	Amit Kumar Sharmaa,∗ , Sandeep Chaurasiaa , Devesh Kumar Srivastavaa [2020]	Sentimental Short Sentences Classification by Using CNN Deep Learning Model with Fine Tuned Word2Vec	Text categorization remains a critical challenge. NLP and deep neural network technologies are now commonly used. Suggested model perfectly cleaned the data, generated word vectors from a pre-trained Word2Vec model, and used a CNN layer to extract improved features for categorizing short phrases.	Kera's is a TensorFlow-based neural network API that uses an embedding layer for real-valued vector representation of words. For the goal of validation accuracy, 9595 sentences or snippets were employed as training sets and 1067 snippets or phrases as testing sets.	Deep learning approaches are particularly effective and efficient for sentiment and semantic analysis. For tiny sentences in the movie review corpus, the research yields a more accurate outcome for feature extraction using Word2Vec and CNN approaches. The proposed method will aid in the classification of text small data from social media and YouTube with more accuracy.

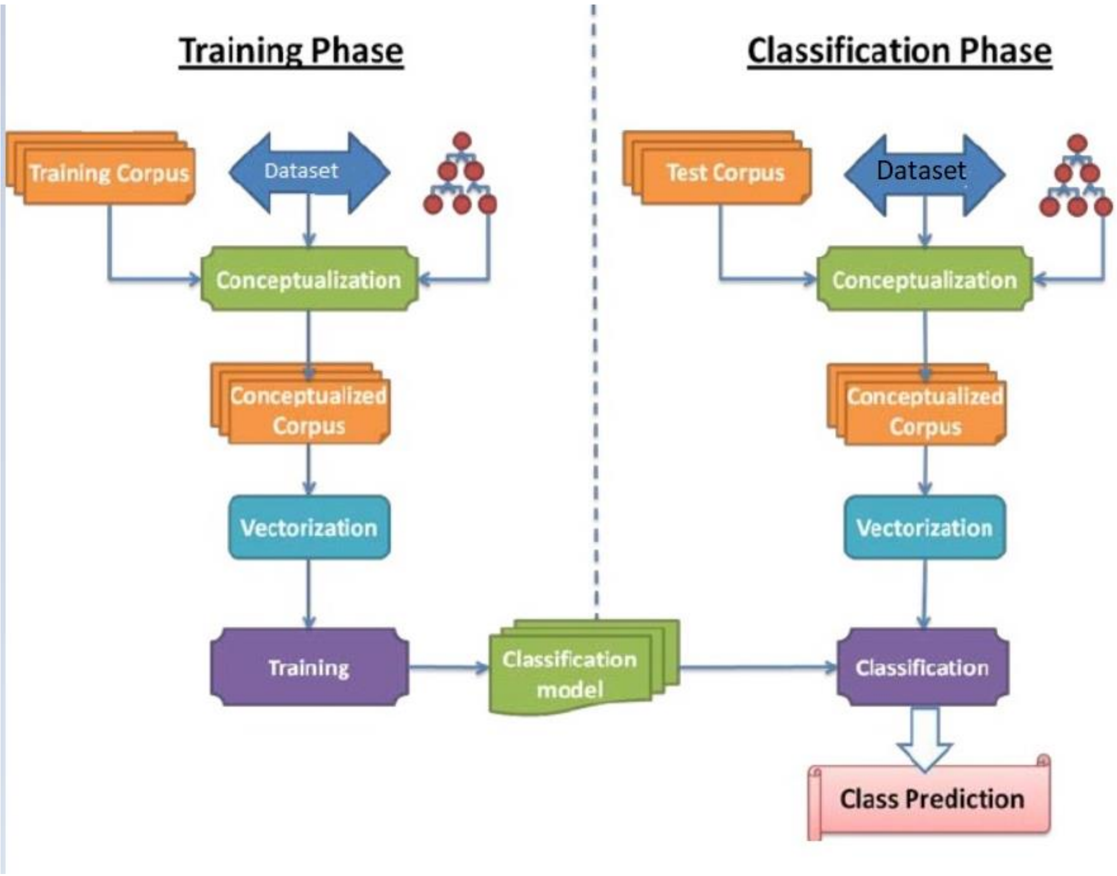
5. PROPOSED SYSTEM BLOCK DIAGRAM

Architecture Diagram:

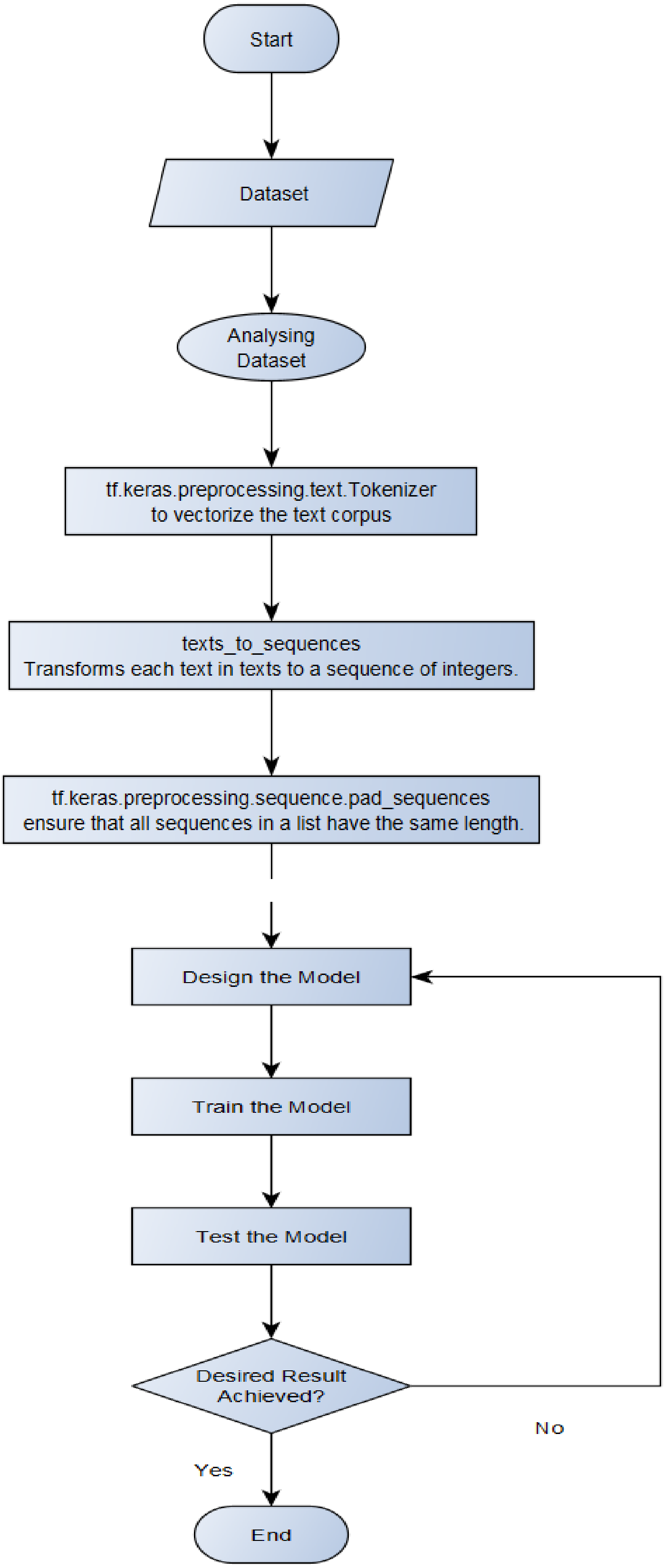
- **Repository Model:**



The following model was chosen as out Dataset is a central entity on which all our functions and preprocessing variables work on and also the model is trained and used on the dataset so as to achieve greater results



Flow Diagram:

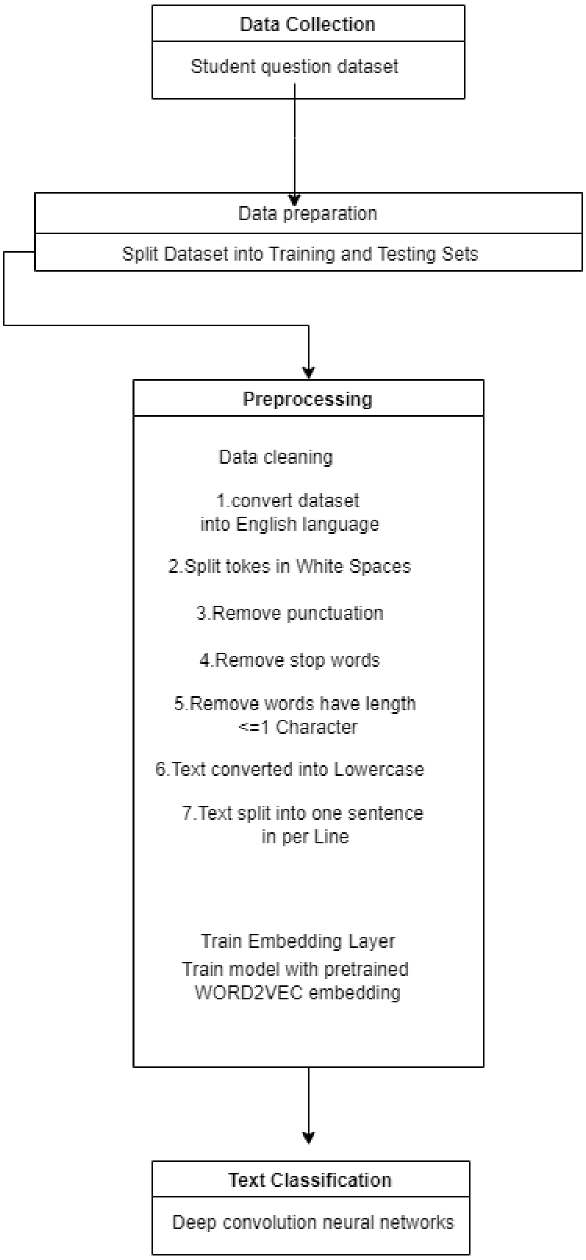

Start
Dataset
Analysing Dataset
tf.keras.preprocessing.text.Tokenizer to vectorize the text corpus
texts_to_sequences Transforms each text in texts to a sequence of integers.
tf.keras.preprocessing.sequence.pad_sequences ensure that all sequences in a list have the same length.
Design the Model
Train the Model
Test the Model
Desired Result Achieved?
No
Yes
End

5.PROPOSED METHODOLOGY AND ANALYSIS

Proposed Model

The proposed working model is based on a deep learning technique for extracting silence features from short utterances.

This approach aids in the classification of subject categorization evaluations. The efficiency of the suggested approach is tested using a dataset of student database. Model that is proposed

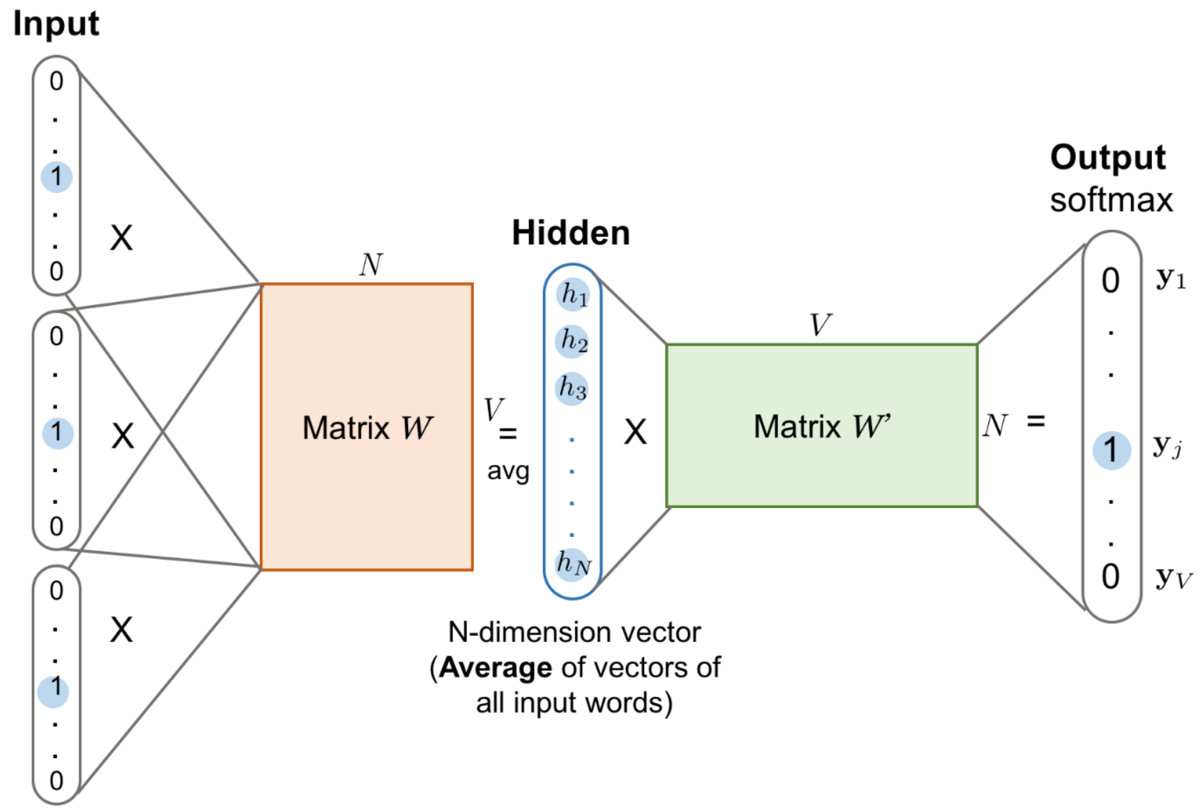

Data Collection
Student question dataset
Data preparation
Split Dataset into Training and Testing Sets
Preprocessing
Data cleaning
1.convert dataset into English language
2.Split tokes in White Spaces
3.Remove punctuation
4.Remove stop words
5.Remove words have length <=1 Character
6.Text converted into Lowercase
7.Text split into one sentence in per Line
Train Embedding Layer
Train model with pretrained WORD2VEC embedding
Text Classification
Deep convolution neural networks

Word2VecModel

For computing continuous distributed representation of words, the Word2Vec model is utilised. This model is a simple neural network model with one hidden layer that provides an efficient implementation of CBOW and Skip-gram architectures for computing vector representations of words. The Word2Vec model outputs word vectors from a text corpus as input. The word vectors are trained using backpropagation and stochastic gradient descent after the algorithm constructs a vocabulary from the input words.

CBOW Architecture:

The final word is derived by averaging the contextual word vectors, and the CBOW architecture predicts contextual words from a given word using a log-linear classifier and Skip-gram architecture trained using a negative sampling approach. Because the context is unrestricted, training examples can be constructed by skipping a certain number of words in the context.



Text Classification using CNN:

The text classification challenge is solved using natural language processing and supervised machine learning.

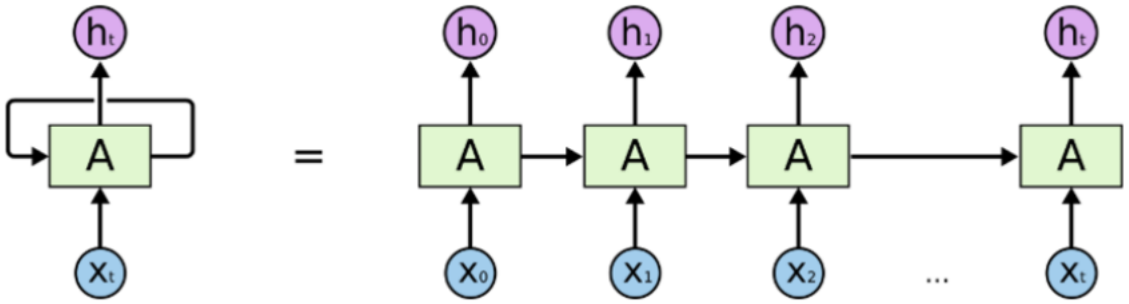
Text classification divides text into one or more prediction categories, such as social media sentiment analysis, ham and spam email identification, customer query auto-tagging, subject prediction, and so on. Convolutional Neural Networks (CNNs) are utilised in this study to extract silent features from sentences using word vectors obtained from the pre-trained Word2Vec model. The CNN layer is in charge of extracting relevant substructures for use in prediction tasks.

CNNs are a type of deep feed-forward neural network that uses multilayer perceptron's to save preparation time. CNN was created to help with picture categorization and computer vision issues. It has recently been used in a variety of NLP tasks. When CNN is used to text rather than images in an NLP task, a one-dimensional array representation of the text is required. The 1D convolutional and pooling operations are part of the CNN architecture.

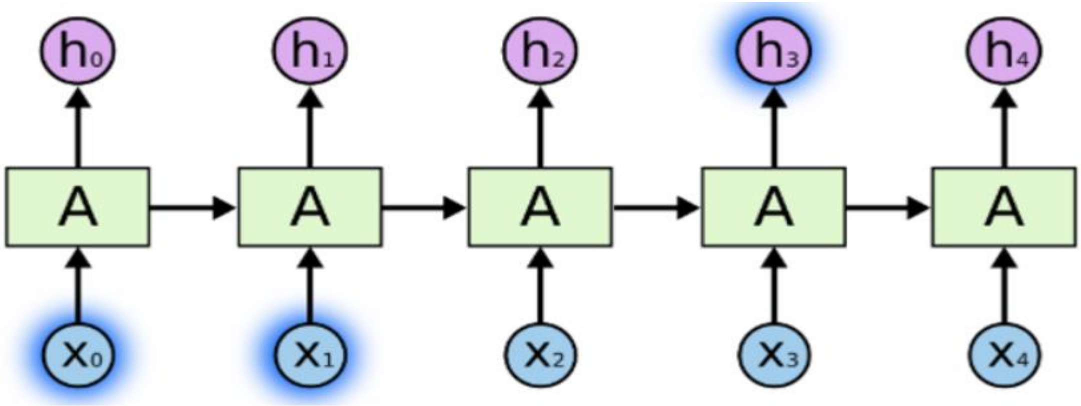
Method description

A lot of innovations on NLP have been how to add context into word vectors. One of the common ways of doing it is using Recurrent Neural Networks. They do use sequential information. They have a memory that picks up what have been calculated so far, i.e. what It spoke over the past will influence what It will speak following. RNNs are ideally suited for text and speech analysis. The most commonly used RNNs are LSTMs.

LSTM (Long Short-Term Memory) network is a type of RNN (Recurrent Neural Network) which is used for learning sequential data prediction problems. Just like any other neural network LSTM also has several layers that help it to learn and identify the pattern for improved performance. Main operation of LSTM can be considered to hold the necessary knowledge and reject the information which is not required or helpful for additional prediction.



‘A’ is one layer of feed-forward neural network. If we only see at the right side, it does recurrently go through the element of each sequence. If we unpack the left, it will look much like the right.



We input each word; words correlate to each other in some ways. By transmitting input from last output, can maintain information, and can leverage all information at the end to make predictions. This work out well for brief sentences, when we deal with a lengthy article, there will be a long-term dependency problem. Thus, we do not usually use vanilla RNNs, and we use Long Short-Term Memory rather. LSTM is a type of RNNs that can resolve this long-term dependency issue. In our text classification for question paper example, we have this many-to-one relationship.

Steps fallowed

- First, we import the libraries we need
  1. Tokenizer: This class allows to vectorize a text corpus, by turning each text into a sequence of integers.
  2. pad sequences: we use pad sequences to make all our questions the same length.
  3. Dense: Dense layer is the usual deeply connected neural network layer. It is most common and frequently used layer. Dense layer does the below operation on the input and return the output. output = activation (dot (input, kernel) + bias).
  4. Input: Input () is used for instantiating Keras tensor. A Keras tensor has a symbolic tensor-like object, which we augment with certain attributes that allow us to build a Keras model just by knowing the inputs and outputs of the model.
  5. GlobalMaxPooling1D: Down samples the input representation by taking the maximum value over the time dimension.
  6. Conv1D: Conv1D class. 1D convolution layer. This layer creates a convolution kernel that is convolved with the layer input over a single spatial dimension to produce a tensor of outputs. If use_bias is True, a bias vector is created and added to the outputs.
  7. MaxPooling1D: Max pooling operation for 1D temporal data. Down samples the input representation by taking the maximum value over a spatial window of size pool_size . The window is shifted by strides.

- Then we create a plot function to project information about the trained model using matplotlib.
- We read the data set using panda's library.
- Change output labeling from (physics, chemistry, math, biology)->(0,1,2,3).
- Now we replace the '\n' line characters to white space to avoid bad input data formation. This is a preprocessing step.
- Split train and test data from data set.
- Use the Tokenizer function to tokenize the train and test input data.
- Use pad_sequences to make all inputs of same maxlen length.
- Create the first neural network architecture using globalmaxpooling1D and dense layers.

- Create the second neural network architecture using LSTM, globalmaxpooling1D and dense layers.
- Create the third neural network architecture using globalmaxpooling1D, Bidirectional and dense layers.
- Create the fourth neural network architecture using Conv1D, globalmaxpooling1D and dense layers.
- Create fifth neural network architecture using Conv1D, GlobalAveragePooling1D, Dropout, Dense. We use dropout layers minimize overfitting.

6.Technology used for text categorization:

Python3 or any latest python IDE (like Anaconda or Jupiter)

Python Libraries Used -

NumPy: It has various mathematical functions for working in the domain of linear

algebra, Fourier transform, and matrices.

pandas: Pandas is an open-source Python package that is most widely used for data science/data analysis and machine learning tasks. here we have used it to extract and pre-process data.

matplotlib: Matplotlib is a cross-platform, data visualization and graphical plotting library for Python and its numerical extension NumPy. As such, it offers a viable open-source alternative to MATLAB. Developers can also use matplotlib's APIs (Application Programming Interfaces) to embed plots in GUI applications.

TensorFlow. keras: Keras is the high-level API of TensorFlow 2: an approachable, highly-productive interface for solving machine learning problems, with a focus on modern deep learning. It provides essential abstractions and building blocks for developing and shipping machine learning solutions with high iteration velocity.

7.Data Interpretation

Because a labelled dataset containing text items and their labels is used to train a classifier, Text Classification is an example of supervised machine learning task. Three primary components make up an end-to-end text classification pipeline:

I. **Dataset Preparation:** The first phase is Dataset Preparation, which entails loading a dataset as well as doing basic pre-processing. After that, the dataset is divided into train and validation sets.

II. **Feature Engineering:** The raw dataset is turned into flat features that may be employed in a machine learning model in the next stage, Feature Engineering. The process of developing new features from existing data is also included in this step.

III. **Model Training:** Finally, a machine learning model is trained on a labelled dataset in the Model Building step.

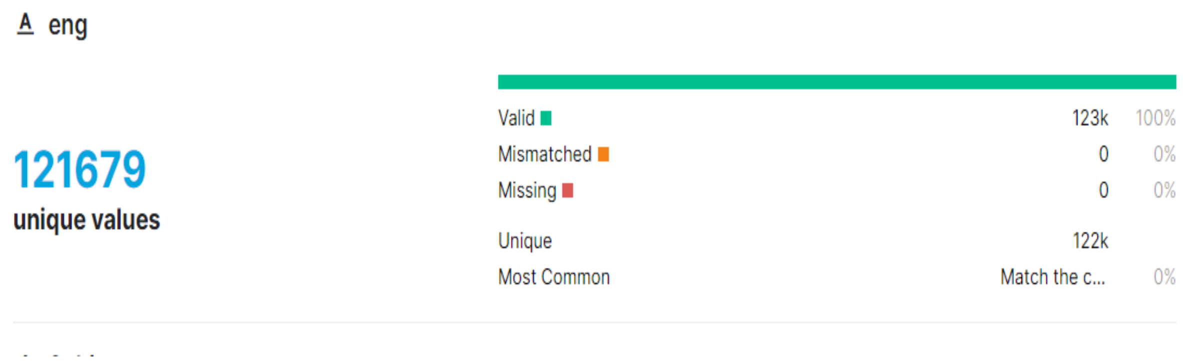
IV. **Improve Text Classifier Performance:** We'll look at a few different strategies to improve text classifier performance in this post.

8.Pseudo-code:

```
Step 1: Start
Step 2: Extract Dataset
Step 3: Vectorize dataset
Step 4: Transform text to sequence of integers
Step 5: Add padding to the texts
Step 6: Design a Model
Step 7: Train the model
Step 8: Check if model is working
Step 9: if
          Desirable result achieved go to Step 9
        Else
          Repeat step 5 onwards
Step 10: End
```

9.Experiments and Results Screenshots:

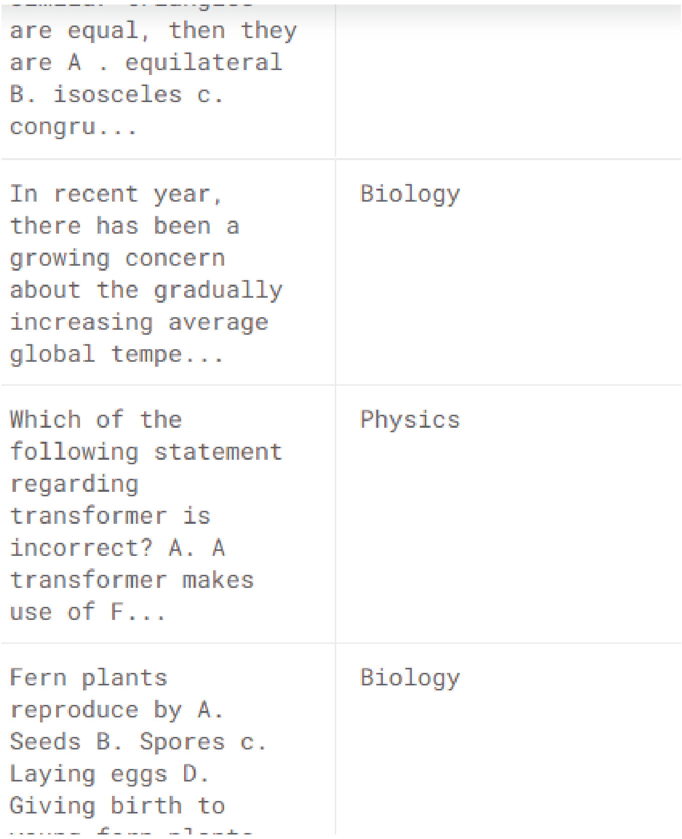
Dataset:



eng

121679
unique values

Valid 🟩	123k	100%
Mismatched 🟧	0	0%
Missing 🟥	0	0%
Unique	122k	
Most Common	Match the c...	0%



are equal, then they are A . equilateral B. isosceles c. congru...	
In recent year, there has been a growing concern about the gradually increasing average global tempe...	Biology
Which of the following statement regarding transformer is incorrect? A. A transformer makes use of F...	Physics
Fern plants reproduce by A. Seeds B. Spores c. Laying eggs D. Giving birth to	Biology

Context

In India, every year lacs of students sit for competitive examinations like JEE Advanced, JEE Mains, NEET, etc. These exams are said to be the gateway to get admission into India's premier Institutes such as IITs, NITs, AIIMS, etc. Keeping in mind that the competition is tough as lacs of students appear for these examinations, there has been an enormous development in Ed Tech Industry in India, fortuning the dreams of lacs of aspirants via providing online as well as offline coaching, mentoring, etc. This particular dataset consists of questions/doubts raised by students preparing for such examinations.

Content

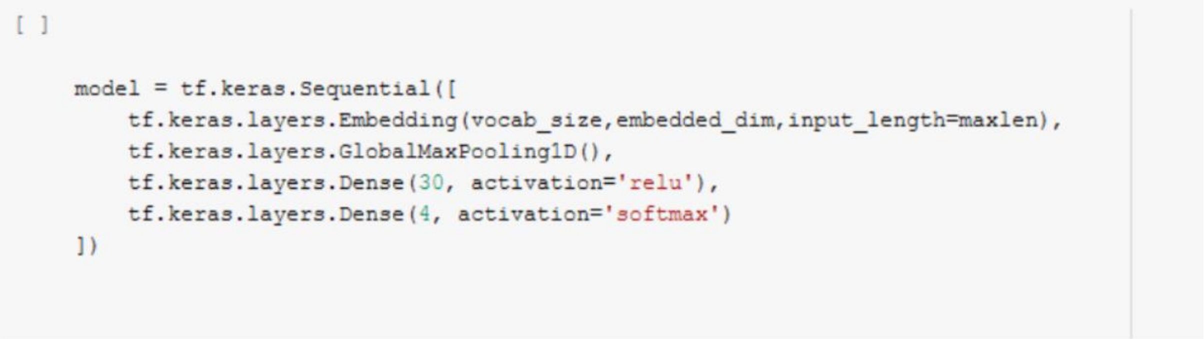
The dataset contains Students-questions.csv file in version 1 as of now.

Inside the CSV file, we have two columns:

eng: The full question or description of the questions

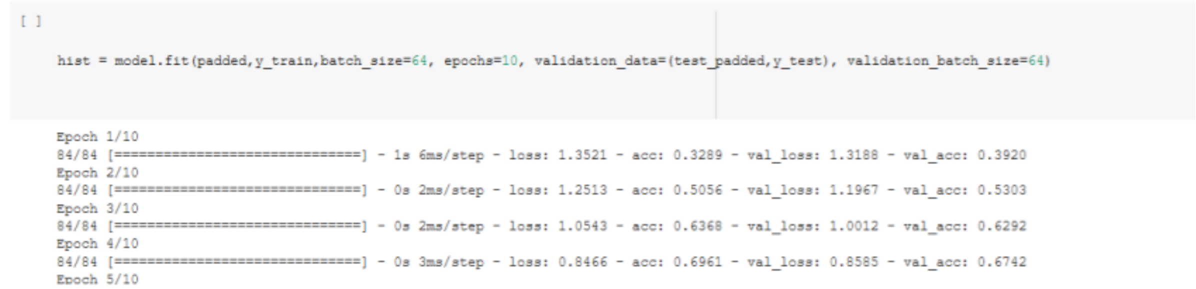
Subject: Which subject does the question belong to. It has 4 classes, Physics, Chemistry, Biology, and Mathematics.

Technique-1:

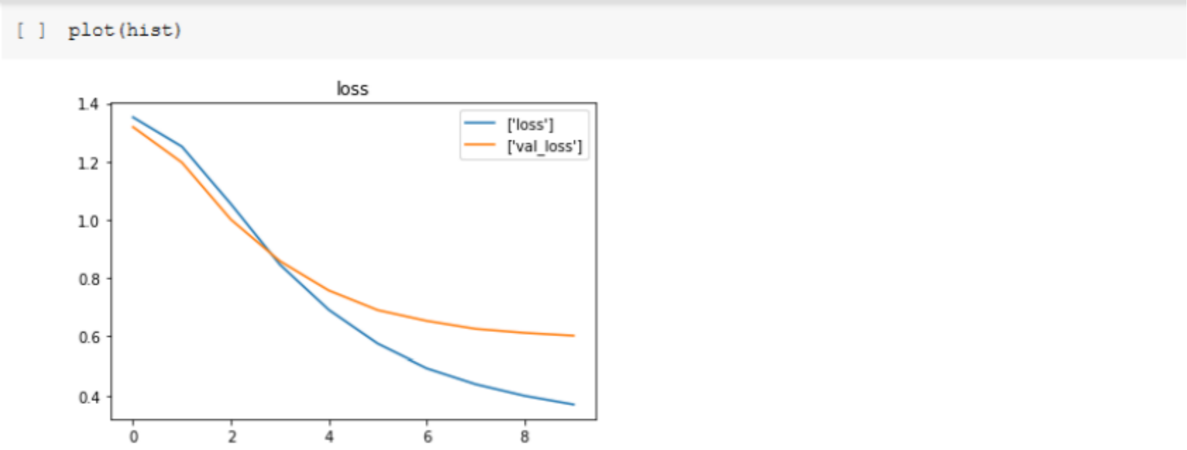


```
[ ]

    model = tf.keras.Sequential([
        tf.keras.layers.Embedding(vocab_size,embedded_dim,input_length=maxlen),
        tf.keras.layers.GlobalMaxPooling1D(),
        tf.keras.layers.Dense(30, activation='relu'),
        tf.keras.layers.Dense(4, activation='softmax')
    ])
```

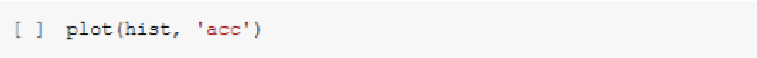
We are using embedded layer for input, globalmaxPolling and dense layer with 30 neurons of relu activation and here we got 0.37 loss and 82 percent accuracy.



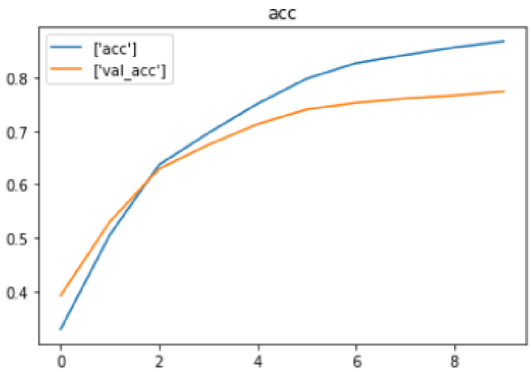
```
[ ]

    hist = model.fit(padded,y_train,batch_size=64, epochs=10, validation_data=(test_padded,y_test), validation_batch_size=64)

    Epoch 1/10
    84/84 [==============================] - 1s 6ms/step - loss: 1.3521 - acc: 0.3289 - val_loss: 1.3188 - val_acc: 0.3920
    Epoch 2/10
    84/84 [==============================] - 0s 2ms/step - loss: 1.2513 - acc: 0.5056 - val_loss: 1.1967 - val_acc: 0.5303
    Epoch 3/10
    84/84 [==============================] - 0s 2ms/step - loss: 1.0543 - acc: 0.6368 - val_loss: 1.0012 - val_acc: 0.6292
    Epoch 4/10
    84/84 [==============================] - 0s 3ms/step - loss: 0.8466 - acc: 0.6961 - val_loss: 0.8585 - val_acc: 0.6742
    Epoch 5/10
```



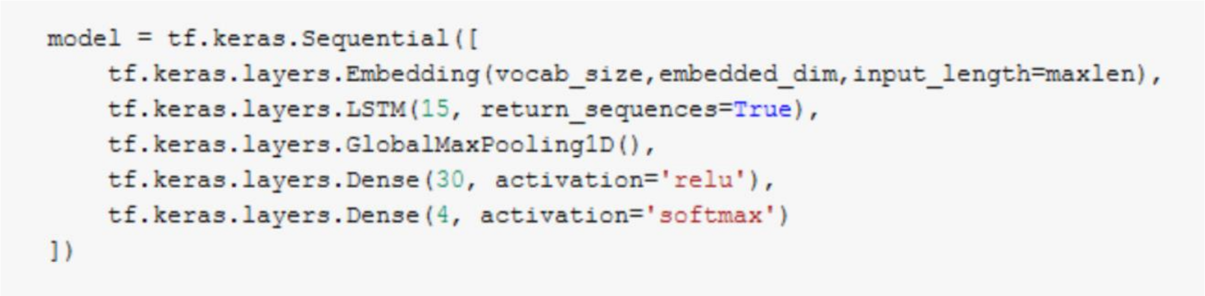
```
[ ] plot(hist)
```



```
[ ]  plot(hist, 'acc')
```

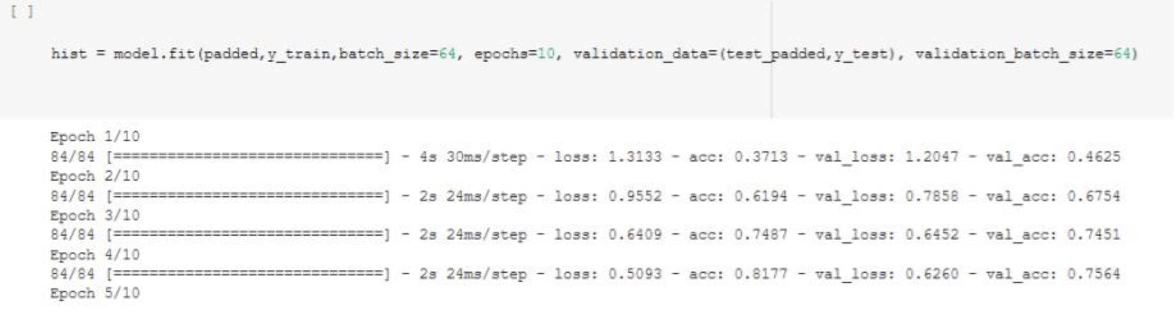


Tehnique-2:

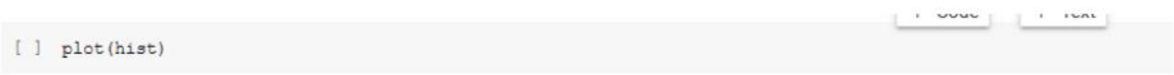


```
model = tf.keras.Sequential([
    tf.keras.layers.Embedding(vocab_size,embedded_dim,input_length=maxlen),
    tf.keras.layers.LSTM(15, return_sequences=True),
    tf.keras.layers.GlobalMaxPooling1D(),
    tf.keras.layers.Dense(30, activation='relu'),
    tf.keras.layers.Dense(4, activation='softmax')
])
```

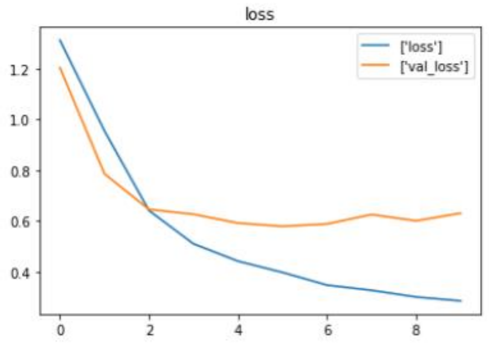
We are using additional LSTM LAYER after input layer followed by, globalmaxPolling and dense layer with 30 neurons of relu activation and here we got 0.48 loss and 84 percent accuracy.

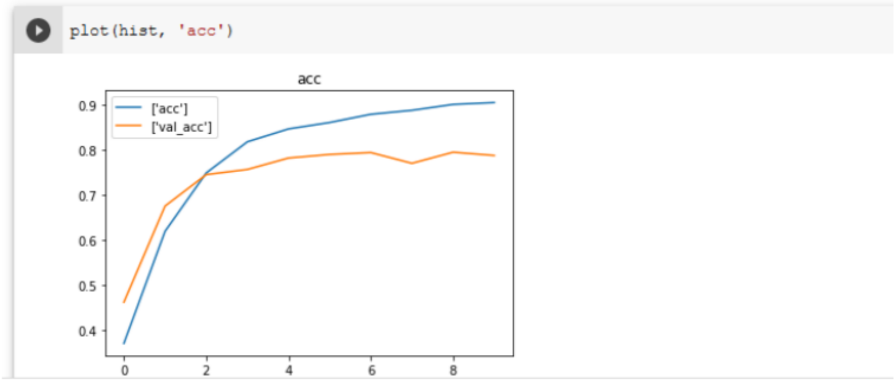


```
[ ]

    hist = model.fit(padded,y_train,batch_size=64, epochs=10, validation_data=(test_padded,y_test), validation_batch_size=64)

    Epoch 1/10
    84/84 [==============================] - 4s 30ms/step - loss: 1.3133 - acc: 0.3713 - val_loss: 1.2047 - val_acc: 0.4625
    Epoch 2/10
    84/84 [==============================] - 2s 24ms/step - loss: 0.9552 - acc: 0.6194 - val_loss: 0.7858 - val_acc: 0.6754
    Epoch 3/10
    84/84 [==============================] - 2s 24ms/step - loss: 0.6409 - acc: 0.7487 - val_loss: 0.6452 - val_acc: 0.7451
    Epoch 4/10
    84/84 [==============================] - 2s 24ms/step - loss: 0.5093 - acc: 0.8177 - val_loss: 0.6260 - val_acc: 0.7564
    Epoch 5/10
```



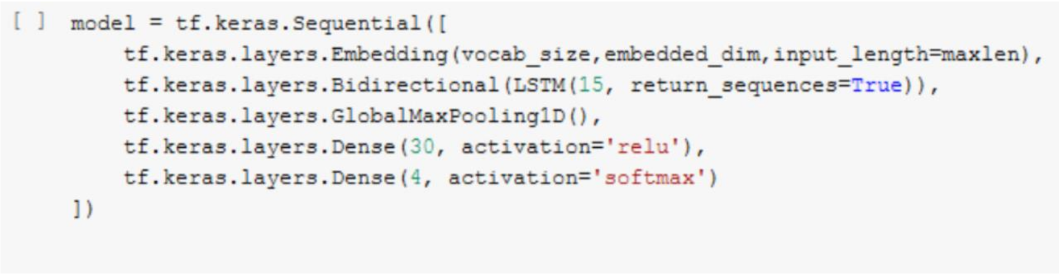
```
[ ]  plot(hist)
```





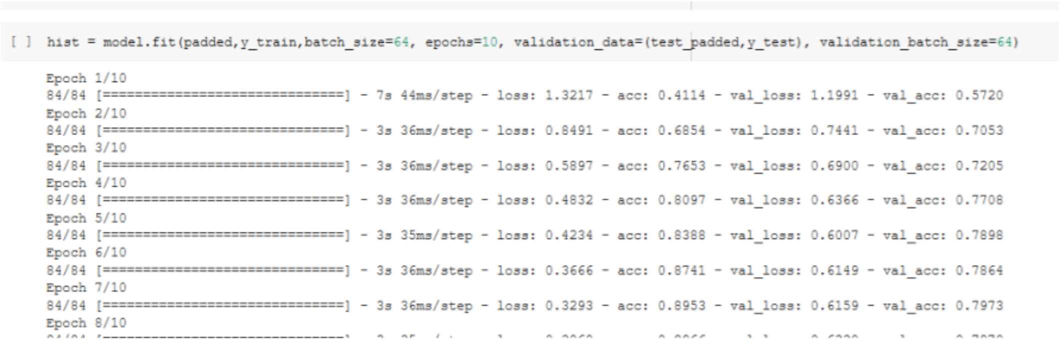
```
plot(hist, 'acc')
```

Technique-3:

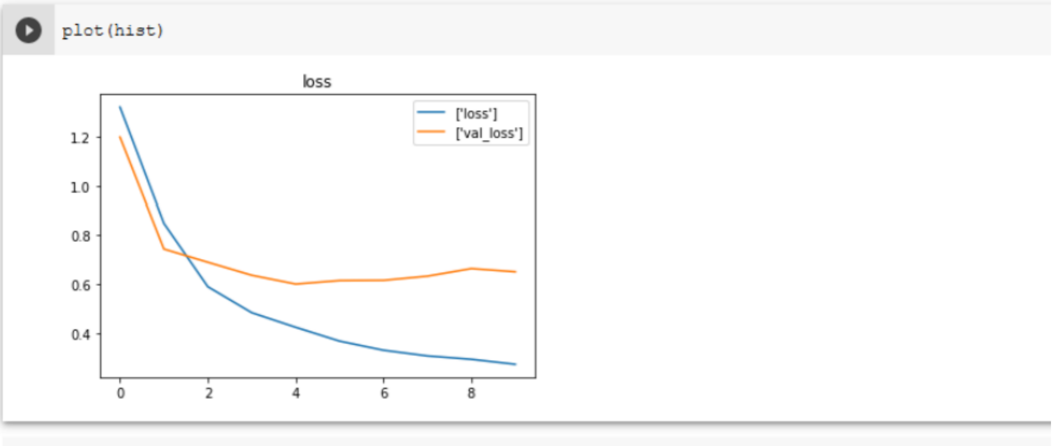


```
[ ]  model = tf.keras.Sequential([
         tf.keras.layers.Embedding(vocab_size,embedded_dim,input_length=maxlen),
         tf.keras.layers.Bidirectional(LSTM(15, return_sequences=True)),
         tf.keras.layers.GlobalMaxPooling1D(),
         tf.keras.layers.Dense(30, activation='relu'),
         tf.keras.layers.Dense(4, activation='softmax')
     ])
```

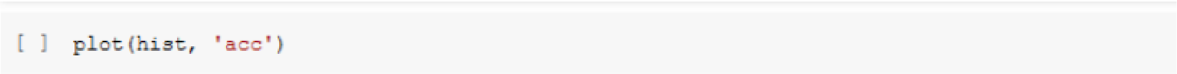
We are using BIDRECTIONAL Wrapped on LSTM LAYER after input layer followed by, globalmaxPolling and dense layer with 30 neurons of relu activation and here we got 0.25 loss and 91 percent accuracy.



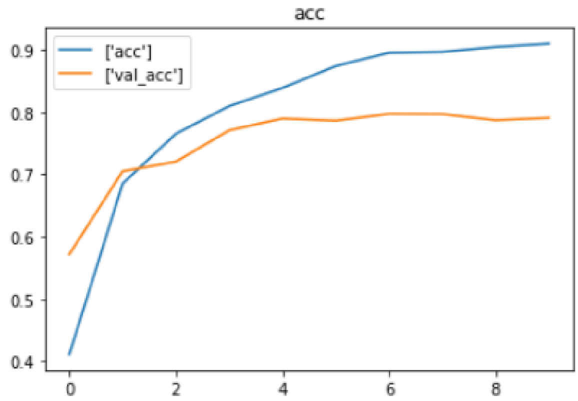
```
[ ]  hist = model.fit(padded,y_train,batch_size=64, epochs=10, validation_data=(test_padded,y_test), validation_batch_size=64)

     Epoch 1/10
     84/84 [==============================] - 7s 44ms/step - loss: 1.3217 - acc: 0.4114 - val_loss: 1.1991 - val_acc: 0.5720
     Epoch 2/10
     84/84 [==============================] - 3s 36ms/step - loss: 0.8491 - acc: 0.6854 - val_loss: 0.7441 - val_acc: 0.7053
     Epoch 3/10
     84/84 [==============================] - 3s 36ms/step - loss: 0.5897 - acc: 0.7653 - val_loss: 0.6900 - val_acc: 0.7205
     Epoch 4/10
     84/84 [==============================] - 3s 36ms/step - loss: 0.4832 - acc: 0.8097 - val_loss: 0.6366 - val_acc: 0.7708
     Epoch 5/10
     84/84 [==============================] - 3s 35ms/step - loss: 0.4234 - acc: 0.8388 - val_loss: 0.6007 - val_acc: 0.7898
     Epoch 6/10
     84/84 [==============================] - 3s 36ms/step - loss: 0.3666 - acc: 0.8741 - val_loss: 0.6149 - val_acc: 0.7864
     Epoch 7/10
     84/84 [==============================] - 3s 36ms/step - loss: 0.3293 - acc: 0.8953 - val_loss: 0.6159 - val_acc: 0.7973
     Epoch 8/10
```



```
plot(hist)
```

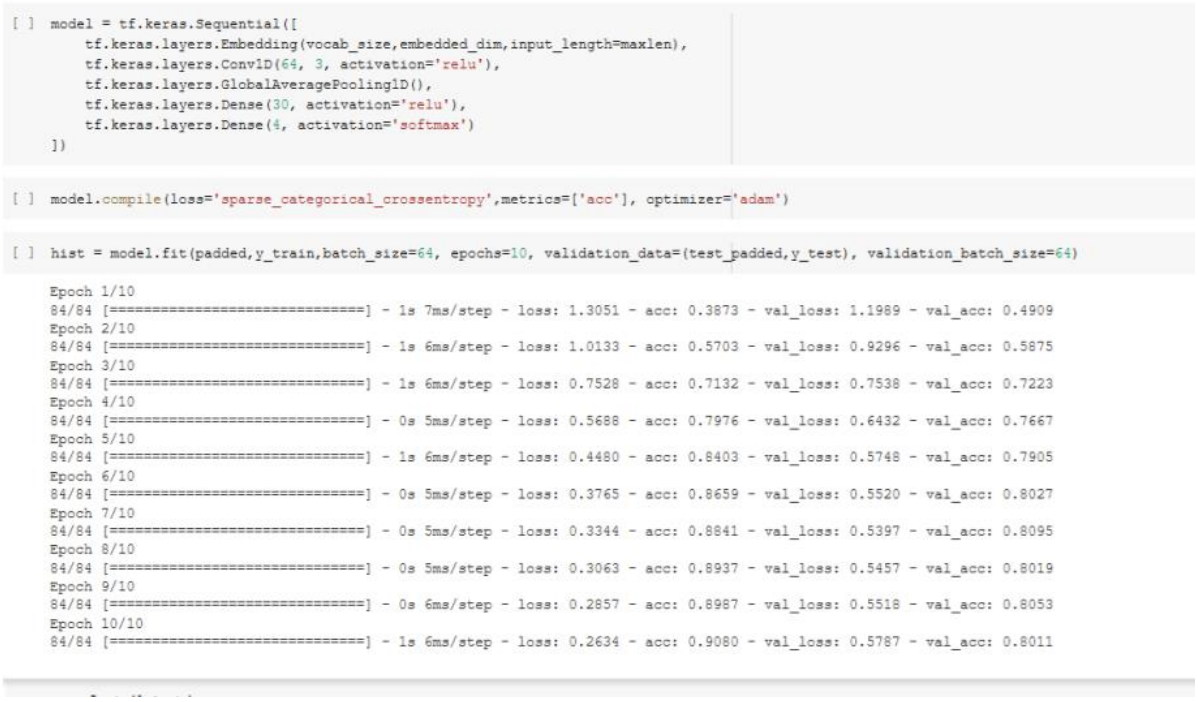


```
[ ]  plot(hist, 'acc')
```



Technique-4:

Here we used CONV1D(64,3) followed by GlobalAveragePolling1D, and dense layer with 30 neurons of relu activation and here we got 0.28 loss and 90 percent accuracy.

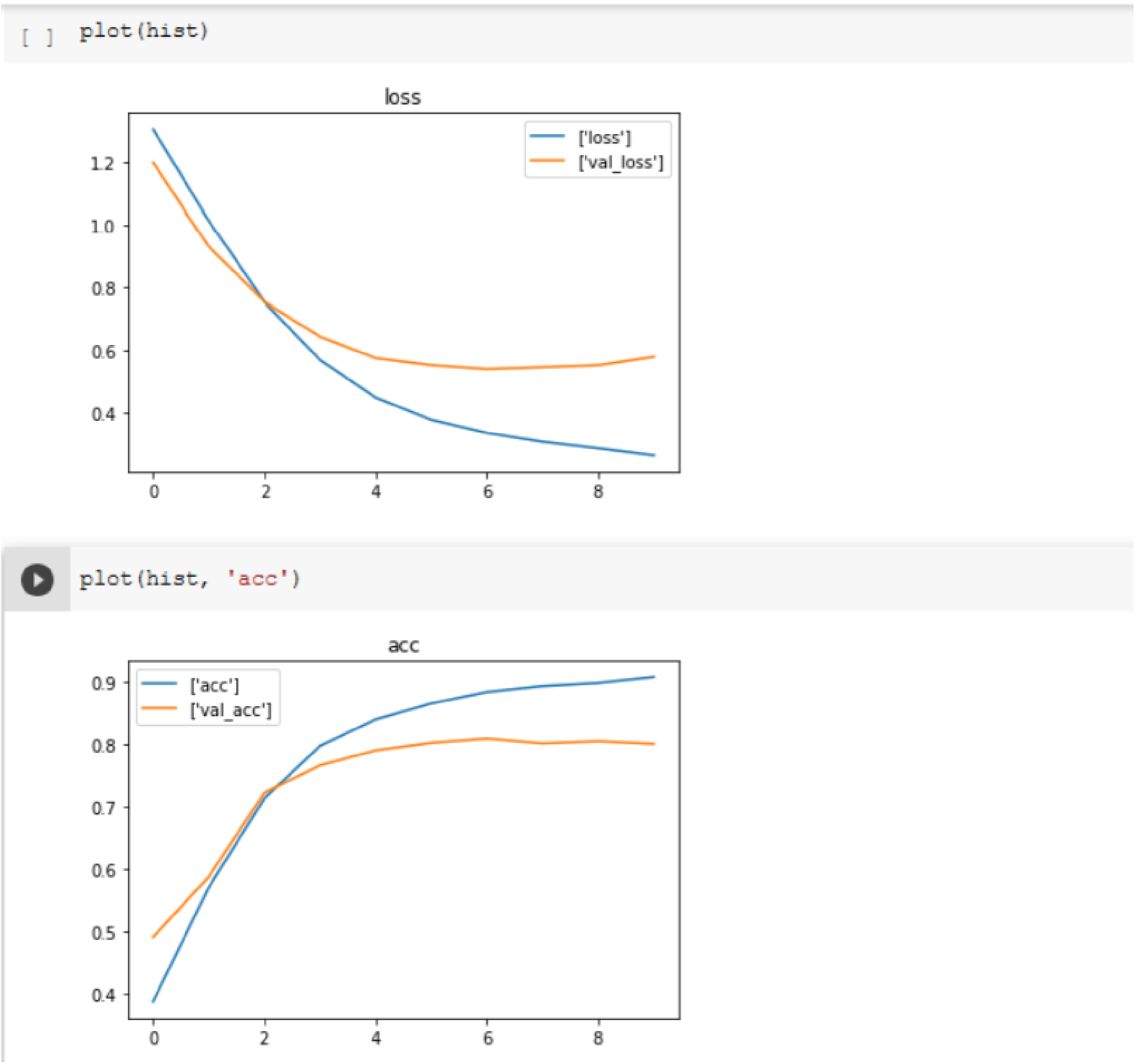


```
[ ]  model = tf.keras.Sequential([
         tf.keras.layers.Embedding(vocab_size,embedded_dim,input_length=maxlen),
         tf.keras.layers.Conv1D(64, 3, activation='relu'),
         tf.keras.layers.GlobalAveragePooling1D(),
         tf.keras.layers.Dense(30, activation='relu'),
         tf.keras.layers.Dense(4, activation='softmax')
     ])
```

```
[ ]  model.compile(loss='sparse_categorical_crossentropy',metrics=['acc'], optimizer='adam')
```

```
[ ]  hist = model.fit(padded,y_train,batch_size=64, epochs=10, validation_data=(test_padded,y_test), validation_batch_size=64)
```

```
Epoch 1/10
84/84 [==============================] - 1s 7ms/step - loss: 1.3051 - acc: 0.3873 - val_loss: 1.1989 - val_acc: 0.4909
Epoch 2/10
84/84 [==============================] - 1s 6ms/step - loss: 1.0133 - acc: 0.5703 - val_loss: 0.9296 - val_acc: 0.5875
Epoch 3/10
84/84 [==============================] - 1s 6ms/step - loss: 0.7528 - acc: 0.7132 - val_loss: 0.7538 - val_acc: 0.7223
Epoch 4/10
84/84 [==============================] - 0s 5ms/step - loss: 0.5688 - acc: 0.7976 - val_loss: 0.6432 - val_acc: 0.7667
Epoch 5/10
84/84 [==============================] - 1s 6ms/step - loss: 0.4480 - acc: 0.8403 - val_loss: 0.5748 - val_acc: 0.7905
Epoch 6/10
84/84 [==============================] - 0s 5ms/step - loss: 0.3765 - acc: 0.8659 - val_loss: 0.5520 - val_acc: 0.8027
Epoch 7/10
84/84 [==============================] - 0s 5ms/step - loss: 0.3344 - acc: 0.8841 - val_loss: 0.5397 - val_acc: 0.8095
Epoch 8/10
84/84 [==============================] - 0s 5ms/step - loss: 0.3063 - acc: 0.8937 - val_loss: 0.5457 - val_acc: 0.8019
Epoch 9/10
84/84 [==============================] - 0s 6ms/step - loss: 0.2857 - acc: 0.8987 - val_loss: 0.5518 - val_acc: 0.8053
Epoch 10/10
84/84 [==============================] - 1s 6ms/step - loss: 0.2634 - acc: 0.9080 - val_loss: 0.5787 - val_acc: 0.8011
```

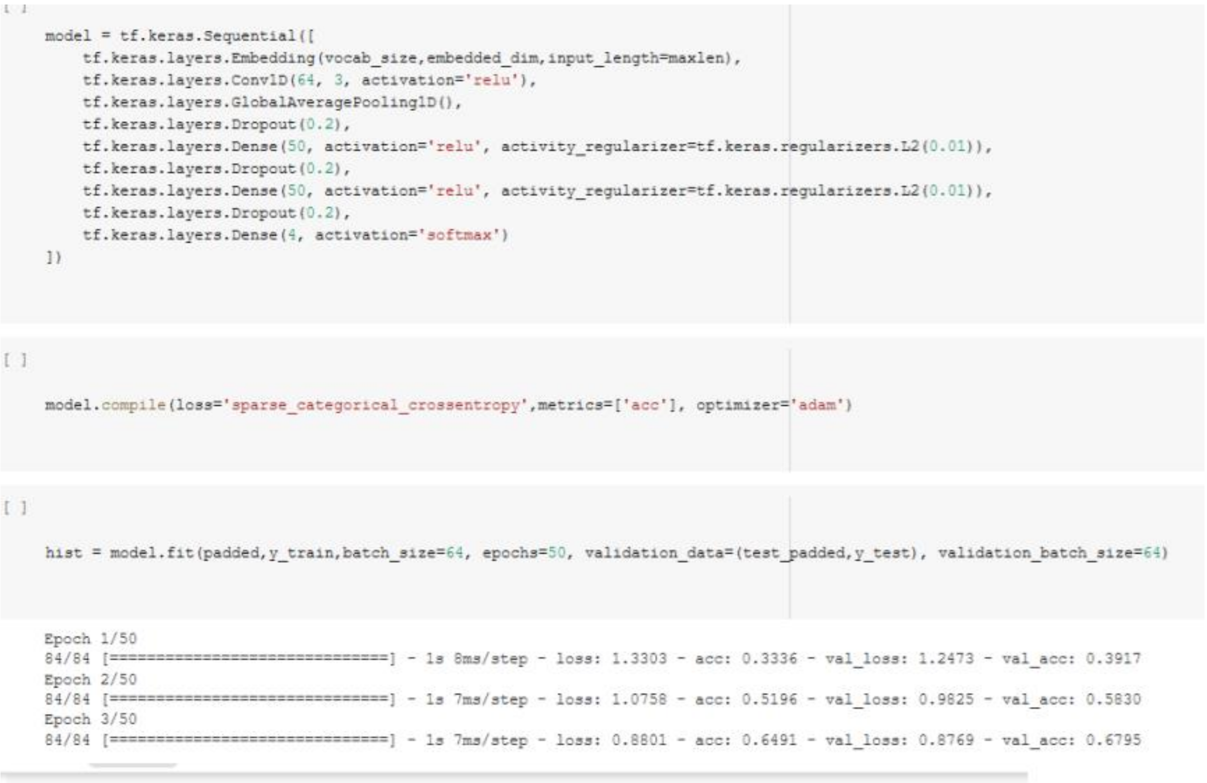


```
[ ]  plot(hist)
```

```
plot(hist, 'acc')
```

Technique-5:

To reduce over fitting and reduce the loss we are using dropout layer where 20 percent data loss, to bring randomization in our model. here we are using dropout layer between CONV1D, GlobalAveragePolling1D to dense layer with 50 neurons RELU Activation. and here we got 0.102 loss and 96 percent accuracy.

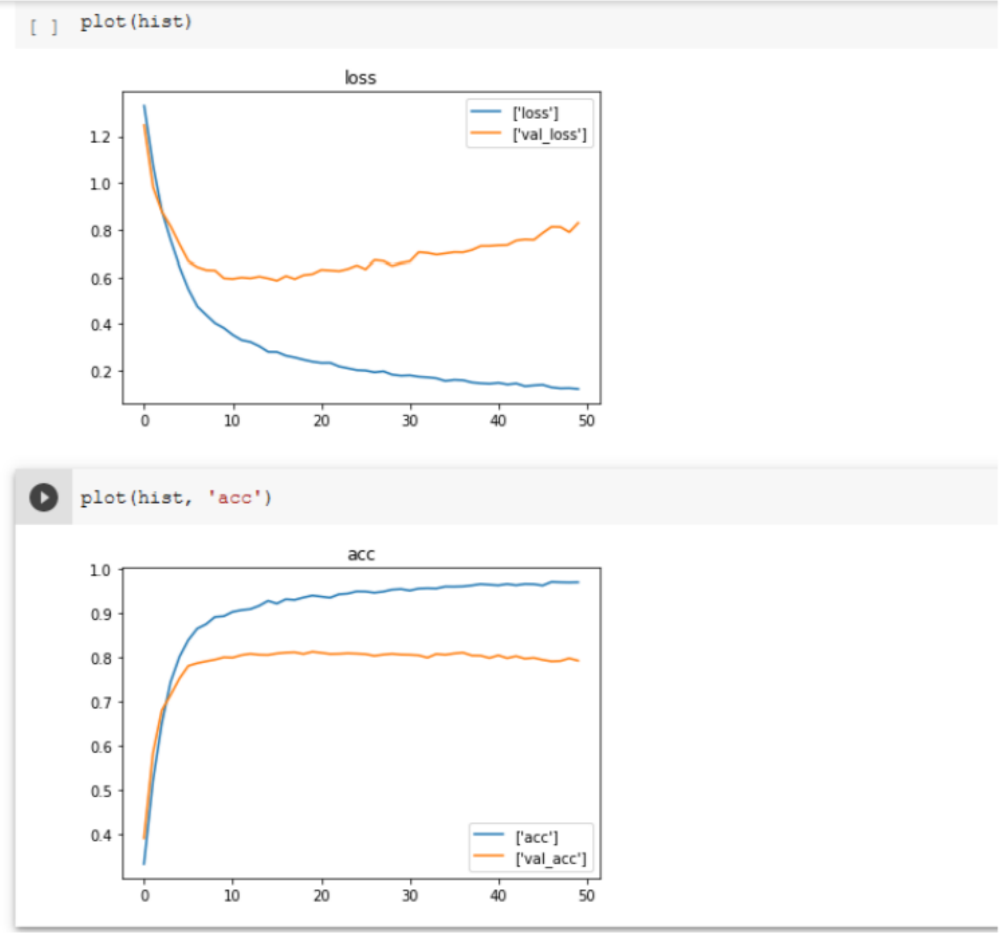


```
[ ]
    model = tf.keras.Sequential([
        tf.keras.layers.Embedding(vocab_size,embedded_dim,input_length=maxlen),
        tf.keras.layers.Conv1D(64, 3, activation='relu'),
        tf.keras.layers.GlobalAveragePooling1D(),
        tf.keras.layers.Dropout(0.2),
        tf.keras.layers.Dense(50, activation='relu', activity_regularizer=tf.keras.regularizers.L2(0.01)),
        tf.keras.layers.Dropout(0.2),
        tf.keras.layers.Dense(50, activation='relu', activity_regularizer=tf.keras.regularizers.L2(0.01)),
        tf.keras.layers.Dropout(0.2),
        tf.keras.layers.Dense(4, activation='softmax')
    ])
```

```
[ ]
    model.compile(loss='sparse_categorical_crossentropy',metrics=['acc'], optimizer='adam')
```

```
[ ]
    hist = model.fit(padded,y_train,batch_size=64, epochs=50, validation_data=(test_padded,y_test), validation_batch_size=64)
```

```
    Epoch 1/50
    84/84 [==============================] - 1s 8ms/step - loss: 1.3303 - acc: 0.3336 - val_loss: 1.2473 - val_acc: 0.3917
    Epoch 2/50
    84/84 [==============================] - 1s 7ms/step - loss: 1.0758 - acc: 0.5196 - val_loss: 0.9825 - val_acc: 0.5830
    Epoch 3/50
    84/84 [==============================] - 1s 7ms/step - loss: 0.8801 - acc: 0.6491 - val_loss: 0.8769 - val_acc: 0.6795
```



```
[ ] plot(hist)
```

```
plot(hist, 'acc')
```

The best technique using dropouts in Neural networks in our project.

10.CONCLUSION:

Based on this literature search, several text classification techniques were identified with their strengths, possibilities and weaknesses in the extraction of knowledge from data. At this stage, it is critical to understand the problems that exist with text classification techniques to facilitate the evaluation of different classifiers. Due to the efficiency of classification, semi-supervised, literature-based text classification is gaining importance in text-mining. Reduce time costs. Some of the other critical issues are performance improvement, handling of large taxonomies, feature selection, document zones, and data imbalance. It is also interesting to conclude that it is not yet practical to prescribe a particular classifier for a particular problem. However, the number of attempts and errors to select the best classifier was minimized based on the information provided in this study. Simpler but powerful algorithms for parameter optimization and streaming data processing are other areas that researchers should explore in the future. It can be concluded that text classification is a well-developed research topic. Simultaneously, it is also a prominent subject in which new approaches can be discovered and the findings can be utilised in various domains. Several issues have not been thoroughly addressed yet. We did not find works explicitly related the problems of over-fitting of text classification models, or transfer, multi-view learning, and dynamic selection classifier, which is the most promising approach for multiple-classifier systems. Moreover, the concept drift which is strong related to data stream analysis requires more research attention.

References:

[1]. A. Basu, C. Walters and M. Shepherd, "Support vector machines for text categorization," 36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the, 2003, pp. 7 pp.-, doi: 10.1109/HICSS.2003.1174243.

[2]. Han EH., Karypis G., Kumar V. (2001) Text Categorization Using Weight Adjusted *k*-Nearest Neighbor Classification. In: Cheung D., Williams G.J., Li Q. (eds) Advances in Knowledge Discovery and Data Mining. PAKDD 2001. Lecture Notes in Computer Science, vol 2035. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-45357-1_9

[3]. X. She and D. Zhang, "Text Classification Based on Hybrid CNN-LSTM Hybrid Model," 2018 11th International Symposium on Computational Intelligence and Design (ISCID), 2018, pp. 185-189, doi: 10.1109/ISCID.2018.10144.
Link: https://ieeexplore.ieee.org/abstract/document/8695507

[4]. Text categorization: past and present

Ankita Dhar1 · Himadri Mukherjee1 · Niladri Sekhar Dash2 · Kaushik Roy

[5]. Supervised and Semi-Supervised Text Categorization using LSTM for Region Embeddings – Rie Johnson and Tong Zhang.
http://proceedings.mlr.press/v48/johnson16.pdf

[6]. Text Categorization and Machine Learning Methods: Current State of the Art – By Durga Bhavani Dasari& Dr. Venu Gopala Rao. K, G. Narayanamma Institute of Technology and Science, Hyderabad
https://computerresearch.org/index.php/computer/article/view/555/555

[7]. P. Semberecki and H. Maciejewski, "Deep learning methods for subject text classification of articles," 2017 Federated Conference on Computer Science and Information Systems (FedCSIS), 2017, pp. 357-360, doi: 10.15439/2017F414.

[8]. W. W. T. Fok et al., "Prediction model for students' future development by deep learning and tensorflow artificial intelligence engine," 2018 4th International Conference on Information Management (ICIM), 2018, pp. 103-106, doi: 10.1109/INFOMAN.2018.8392818.

[9]. Abu, Mohd Azlan & Indra, Nurul Hazirah & Abd Rahman, Abdul & Sapiee, Nor & Ahmad, Izanoordina. (2019). A study on Image Classification based on Deep Learning and Tensorflow. 12. 563-569.

[10]. F. Miao, P. Zhang, L. Jin and H. Wu, "Chinese News Text Classification Based on Machine Learning Algorithm," 2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2018, pp. 48-51, doi: 10.1109/IHMSC.2018.10117.

[11]. Prasanna, P. & Rao, Dr. (2018). Text classification using artificial neural networks. International Journal of Engineering and Technology(UAE). 7. 603-606. 10.14419/ijet.v7i1.1.10785.

[12]. Sentimental Short Sentences Classification by Using CNN Deep Learning Model with Fine Tuned Word2Vec

Amit Kumar Sharmaa,∗ , Sandeep Chaurasiaa , Devesh Kumar Srivastavaa
https://www.sciencedirect.com/science/article/pii/S1877050920308826

[13]. High Accuracy Rule-based Question Classification using Question Syntax and Semantics. Harish Tayyar Madabushi, Mark Lee
https://aclanthology.org/P16-1116/

[14]. Question Classification using HDAG Kernel, Jun Suzuki, Hirotoshi Taira, Yutaka Sasaki, and Eisaku Maeda.
https://aclanthology.org/W03-1208.pdf

[15]. Dell Zhang and Wee Sun Lee. 2003. Question classification using support vector machines. In Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval (SIGIR '03). Association for Computing Machinery, New York, NY, USA, 26–32. DOI:https://doi.org/10.1145/860435.860443