ISTANBUL COMMERCE UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

A METHOD FOR PREDICTING TITLE OF GIVEN TEXT

Mohamed Barre OMER

Supervisor
Assoc.Prof. Dr. Mustafa Cem KASAPBAŞI

MASTER'S THESIS
DEPARTMENT OF COMPUTER ENGINEERING
ISTANBUL-2021

ACCEPTANCE AND APPROVAL PAGE

On 06/10/2021 **Mohamed Barre OMER** successfully defended the thesis, entitled "**A Method for Predicting Title of Given Text**" which he prepared after fulfilling the requirements specified in the associated legislation, before the jury members whose signatures are listed below. This thesis is accepted as a **Master's Thesis** by İstanbul Commercial University, Graduate School of Natural and Applied Science **Department of Computer Engineering**.

Supervisor	**Assoc. Prof. Dr. Mustafa Cem KASAPBAŞI** Istanbul Commerce University	.....
Jury Member	**Asst. Prof. Dr. Arzu KAKIŞIM** Istanbul Commerce University	.....
Jury Member	**Assoc. Prof. Dr. Buket DOĞAN** Marmara University	.....

Approval Date: 03.12.2021

Istanbul Commerce University, Graduate School of Natural and Applied Sciences, accordance with the 2nd article of the Board of Directors Decision dated 03/12/2021 and numbered 2021/329, "Mohamed Barre OMER" who has determined to fulfill the course load and thesis obligation was unanimously decided to graduated.

Prof. Dr. Necip ŞİMŞEK
Head of Graduate School of Natural and Applied Science

DECLERATION OF CONFORMITY IN ACADEMIC AND ETHIC CODES

In this thesis study, which I prepared in accordance with the thesis writing rules of Istanbul Commerce University, Institute of Science and Technology,

- I have obtained all the information and documents in the thesis within the framework of academic rules,
- I present all visual, audio and written information and results in accordance with scientific ethics,
- In case of benefiting from the works of others, I refer to the relevant works in accordance with scientific norms,
- I show all the works I refer to as a source,
- I have not made any falsifications in the data used,
- and that I have not submitted any part of this thesis as another thesis at this university or another university.

I declare.

03/12/2021

Mohamed Barre OMER

CONTENTS

Page

CONTENTS i
ABSTRACT iii
ÖZET iv
ACKNOWLEDGEMENTS v
TABLE OF FIGURES vi
LIST OF TABLES vii
LIST OF ABBREVIATION WORDS viii
1.INTRODUCTION 1
1.1.Motivation of the Study 1
1.2 Brief introduction to Summarization 2
1.3 Summarization Approaches 2
2.LITERATURE REVIEW 3
2 .1 NLP (Natural Language Processing) Applications 3
2.1.1 Information extraction 3
2.1.2 Sentiment analysis 4
2.1.3Opinion summarization 4
2.1.4 Speech recognition 5
2.1.5 Other application of NLP 5
2.2 Preparing data 5
2.2.1 Data preprocessing 5
2.2.2 Word embedding. 5
2.3 Brief Scientific Background Information About Text Generation 8
2.4 Attention Mechanism 9
3.METHODOLOGY 10
3.1 Model diagram 10
3.2 Long short memory (LSTM) 10
3.3 Algorithm Steps 11
3.3.1 Data set reading 11
3.3.2 Preprocessing data 12
3.3.3 Check if data cleaned if yes then go next step else back preprocess data 12
3.4.4 Encoder decoder with LSTM 12
3.3.5 Train model 12
3.3.6 Summary generation 12
3.3.7 Long Short Term Memory (LSTM) 13
3.4 Sequence to Sequence 13
3.4.1 Language model 13
3.5 Dataset and Training 14
3.6 Using Google Colaboratory 15
3.7 Lunching Google Colab 15
3.8 Benefits of Google Colaboratory 16
4. EVALUATION RESULTS AND DISCUSSIONS 17
4.1 Rouge 17
4.2 Results and Discussions 19
5. CONCLUSION AND IMPLICATIONS 21
REFERENCES 22

APPENDIX 24
BIOGRAPHY 28

ABSTRACT

M.Sc. Thesis

A METHOD FOR PREDICTING TITLE OF GIVEN TEXT

Mohamed Barre OMER

İstanbul Commerce University
Graduate School of Applied and Natural Sciences
Department of Computer Engineering

Supervisor: Assoc. Prof. Dr. Mustafa Cem KASAPBAŞI

2021, 28 pages

Nowadays, tremendous text data resources are everywhere in the form of books, news journals, websites, and many more. Exploring and gaining insight into text data is very crucial. The titles give a summary of the article and history, getting a coherent semantically, and syntactically title is quite a challenging task. In this study, a deep learning system namely the LSTM (Long Short-Term Memory) neural system is proposed for predicting the title of a given text (PTT) which is a natural language generation system. Deep neural network architecture recently gains popularity, which is easier than previous statistical models for generating text. In this study publicly open news dataset is used from Kaggle called news summary for headline generation. A 500 hundred news summary subset is chosen out of 98403 records for efficiency and less processing power requirements. Firstly, stop words are removed as preprocessing then punctuations are corrected and text is transformed to lower case. Later Porter Stemmer is used to obtaining stems of the words in the text. After tokenization, it is divided into 16-word-length sequences. In order to feed numerical values to LSTM word embedding is utilized. The proposed LSTM model generated high-quality titles according to human evaluation based on results we get from Recall-Oriented Understanding for Gisting Evaluation (ROUGE) as for ROUGE 1 Average_ Recall: 0,69886, Average Precision :0,69924, Average_F1:0,69905 as for ROUGE 2 Average_Recall:0,69874, Average Precision :0,69895, Average_F1:0,69884, as for ROUGE L Average_Recall:0,69829, Average_Precision:0,69829 Average_F1:0,69829.

Keywords: Deep learning, LSTM, NLP, rouge, text generation.

ÖZET

Yüksek Lisans Tezi

VERİLEN METNİN BAŞLIĞINI TAHMİN ETMEK İÇİN BİR YÖNTEM

Mohamed Barre OMER

İstanbul Ticaret Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Mustafa Cem KASAPBAŞI

2021, 28 pages

Günümüzde, muazzam metin veri kaynakları her yerde kitaplar, haber dergileri, web siteleri ve çok daha fazlası şeklindedir. Metin verilerini keşfetmek ve bunlarla ilgili içgörü kazanmak çok önemlidir. Başlıklar makalenin ve tarihin bir özetini verir, anlamsal ve sözdizimsel olarak tutarlı bir başlık elde etmek oldukça zor bir iştir. Bu çalışmada, doğal bir dil üretme sistemi olan belirli bir metnin (PTT) başlığını tahmin etmek için LSTM (Long Short Term Memory) sinir sistemi adlı bir derin öğrenme sistemi önerilmiştir. Derin sinir ağı mimarisi son zamanlarda popülerlik kazanıyor ve bu, metin oluşturmak için önceki istatistiksel modellerden daha kolay. Bu çalışmada, Kaggle'ın haber özeti adı verilen halka açık haber veri kümesi kullanılmıştır. Verimlilik ve daha az işlem gücü gereksinimleri için 98403 kayıt arasından 500 yüz haber özeti alt kümesi seçilir. Önce durak sözcükleri ön işleme olarak kaldırılır, ardından noktalama işaretleri düzeltilir ve metin küçük harfe dönüştürülür. Daha sonra Porter Stemmer, metindeki kelimelerin köklerini elde etmek için kullanılmıştır. Tokenizasyondan sonra, 16 kelimelik dizilere bölünür. Sayısal değerleri LSTM'ye beslemek için kelime gömme işlemi kullanılır. Önerilen LSTM modeli, ROUGE için Rough Recall Oriented (ROUGE)'dan aldığımız sonuçlara göre ROUGE 1 Ortalama_ Recall: 0,69886, Ortalama_Presicion:0,99924, Ortalama_F1:0,69905 için insan değerlendirmesine göre yüksek kaliteli başlıklar üretti. 2 Ortalama_Hatırlatma:0,69874, Ortalama_Hassas :0,69895, Ortalama_F1:0,69884, ROUGE L Ortalama_Geri Çağırma:0,69829, Ortalama_Hassas:0,69829 Ortalama_F1:0,69829.

Anahtar Kelimeler: Deep learning, LSTM, NLP, rouge, text generation.

ACKNOWLEDGEMENTS

I would like to thank to my supervisor, Assoc. Prof. Dr. Mustafa Cem KASAPBAŞI for his countless helping to overcome the difficulties I faced during my thesis study and also for his guidance and advices throughout this research, and my family with their countless moral support.

Mohamed Barre OMER
İstanbul, 2021

TABLE OF FIGURES

Page

Figure 1.1 Biological neuron versus artificial neural network 1
Figure 2.1 Relationship between NLP and ML 3
Figure 2.2 Information extraction architecture pipeline 4
Figure 2.3 Text cleaning architecture 6
Figure 2.4 Word embeddings 7
Figure 2.5 Long short-term memory (LSTM) with attention model 9
Figure 3.1 Model diagram 10
Figure 3.2 Long short time memory (LSTM) 11
Figure 3.3 Sequence to sequence model architecture 12
Figure 3.4 TESLA K80 vs CPU 13
Figure 3.5. Google colab home page 15
Figure 3.6 Python notebook 16
Figure 4.1 System predicted titles 19
Figure 4.2 Accuracy and los during training 20

LIST OF TABLES

Page

Table 4.1 Average rouge scores 18

LIST OF ABBREVIATION WORDS

ATS	Automatic Text Summarization
BLUE	Bilingual Understanding Evaluation
CNN	Convolutional Neural Network
LSTM	Long Short-Term Memory
NER	Named Entity Recognition
NLP	Natural language processing
PTT	Prediction Title of Text
ROUGE	Recall Oriented Understanding Gisting Evaluation
TF-IDF	Text frequency inverse Document

1. INTRODUCTION

1.1. Motivation of the Study

As the quantity of unstructured text data increases, applications that automatically summarize documents gain more popularity and become important. It saves time and effort for finding useful information.

Russell, (2018) defined machine learning as practice of programming computers to learn from data deep learning is subfield of machine learning see figure 2.1 Relationship between NLP and ML.

Machine learning is a subfield of artificial intelligence that gives computers the ability to learn without explicitly being programmed (Sara, 2021)

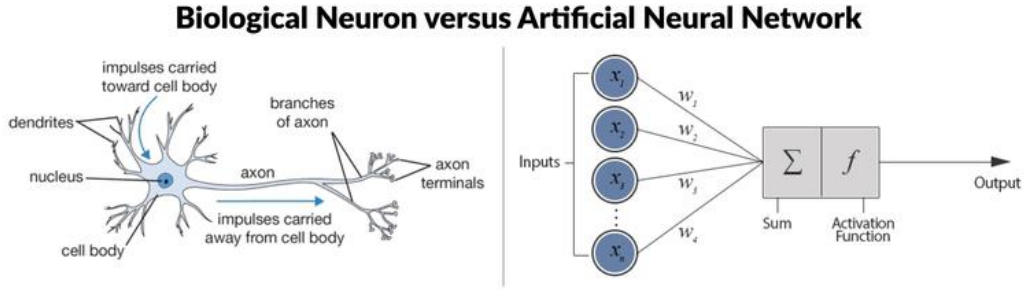

Figure 1.1 Biological neuron versus artificial neural network (Leung,2019)

Jain et al., (1996) A neuron (sometimes known as a nerve cell) is a type of biological cell that processes data (see Figure 1). The axon and dendrites are two types of out-reaching tree-like branches that make up the cell body, or soma. The cell body has a nucleus that stores information about inherited features and plasma that includes the molecular machinery for producing the material that the neuron needs. A neuron's dendrites (receivers) receive signals (impulses) from other neurons and transmits signals generated by its cell body via the axon (transmitter), which eventually forks into strands and sub strands. Synapses are located at the ends of these strands. A synapse is a basic structure.

If you look at Figure 1.1 neural network verses biological network neural network model derived from biological model, Artificial neural network has input layer, hidden layer and out put layer. For example x1, x2, x3…xn are number of inputs n, w1,w2,w3..wn are the weights ,sigma is summation unit ,the function is threshold then out put .

1.2 Brief Introduction to Summarization

Summarization is the process of compressing a portion of the Text to a shorter version that includes the main points of original documents(See et al., 2017) Headlines generation is subfield of text summarization it produces a concise and salient shorter document from large document.

1.3 Summarization Approaches

There are two approaches for text summarization: *extractive* and *abstractive* extractive summary select most important sentences of document and joined it is subset of original text; while in abstractive summary new sentences are generated and paraphrased. (Allahyari et al., 2017)

2. LITERATURE REVIEW

Before giving scientific background studies comparatively about title generations, some background information about Natural Language Processing will be given.

2 .1 NLP (Natural Language Processing) Applications

Natural Language Processing is subfield of AI (Artificial Intelligence). Some of the applications of NLP are Information extraction, summarization, sentiment analysis, text classification, topic segmentation, question answering, and speech recognition.(Mehryar Mohri Afshin, 2018).

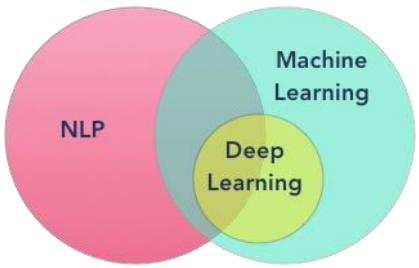

Figure2.1 Relationship between NLP and ML (Framton, 2018)

2.1.1 Information extraction

Information extraction is the process of obtaining meaningful information out of given text in automated manner. ((Chopra et al., 2016)). In figure 1.3 Information extraction architecture is given. According to this architecture first step is sentence segmentation of raw text. In this step long sentences are segmented into words

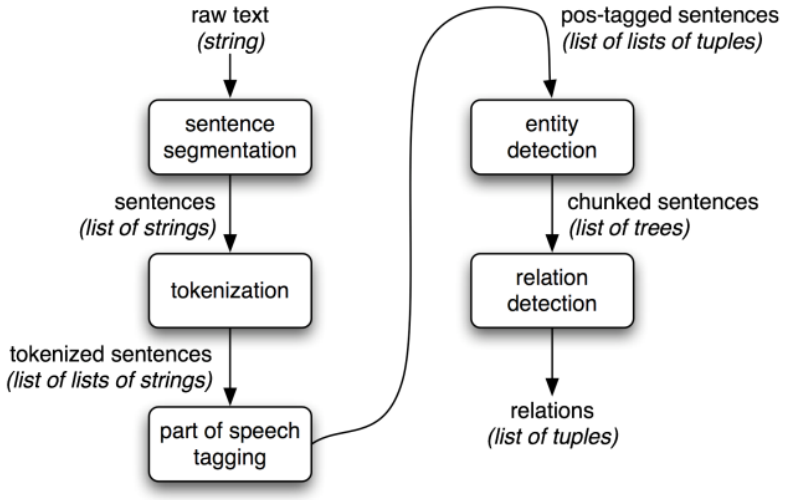

Figure2.2: Information extraction architecture pipeline (NLTK, 2021)

2.1.2 Sentiment analysis

Sentiment analysis refers to analyzing an opinion or feelings about something using data like text or images, regarding almost anything Sentiment analysis is the process of deducting positive or negative sentiments. It's used by businesses to understand brand reputation; measure sentiments in social data as a result of customers are understood better. For example, analyzing automatically thousands of reviews about a product can help businesses to understand better customers are happy about product or service.

2.1.3 Opinion summarization

Opinions, unlike factual data, are ultimately subjective. A single point of view from a single person is generally insufficient to take action. In most cases, it is necessary to study the viewpoints of a significant number of people. This suggests that a summary of viewpoints is required. Although an opinion summary can take numerous forms, such as a structured summary (see below) or a short text summary, the key components of a summary should include perspectives on various entities and their characteristics, as well as a quantitative perspective. The quantitative perspective is very significant since 20% of people being happy about a product is considerably different from 80% of people being positive about a product. (Lin et al., 2016)

2.1.4 Speech recognition

Speech recognition is subfield of computer science and computational linguistics that enables to develop methodologies to recognize and translation of spoken language into text(Trilla, 2009)

2.1.5 Other application of NLP

There are more many applications in NLP one example of them is Question answering which is speech to text applications which contribute many benefits to students with special needs

2.2 Preparing Data

Data Preparation for Machine Learning You will be able to examine, clean, and organize your data in ways that help your machine learning model perform better First, the following steps discover why data cleaning and preparation are so crucial, as well as how to deal with missing data, outliers, and other data-related issues.

2.2.1 Data preprocessing

Data Preprocessing is an important procedure that is use to convert crude data into more clean data in order to be suitable for further analysis. Some of Data Preprocessing steps take in this work is:

a. Tokenization which is breaking down of sentence into words since it is needed individual entities to workable.
b. Stop word removal Stop words these are common a word that doesn't hold much value in dataset also doesn't contribute in analyzing process; such common word was removed as well as punctuations.

c. Stemming is to reduce the word into the word stem back to their roots, in order to avoid words with similar meaning to be considered as different, played, playing was reduced to its root word play, by doing so it to have easier way to analyze of data. However not all time is suitable to apply stemming because it sometime does unnecessary stemming such it reduces the ring to r which will not have any meaning, also lemmatization was considered, it is important algorithmic process of grouping together different infected of words in order to analyzed as single.

d. Converting to lower case; The upper-case word was converted to lower case as part data preprocessing it help to get rid of unhelpful parts of the data.

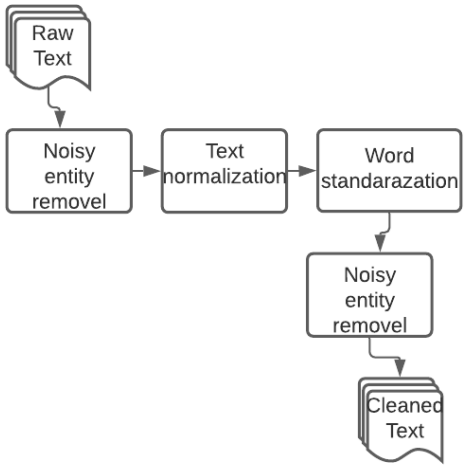

Figure 2.3 Text cleaning architecture

2.2.2 Word embedding

Word embeddings are a set of models and approaches for converting words (or phrases) into vector space and displaying them in a high-dimensional field. You can use this information to determine the degree of similarity (or dissimilarity) between two words (or phrases, or documents).(Hermawan, 2011; Google, 2021) figure 2.4 below shows word embedding

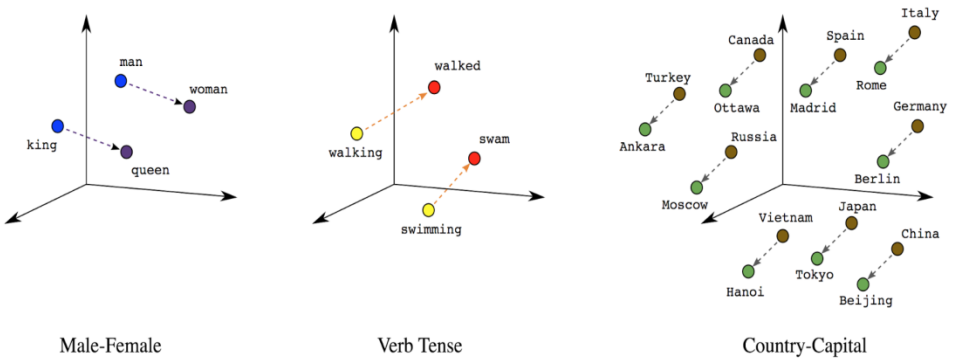

Figure 2.4 Word embeddings

Boehmke and Greenwell, (2019) most state of art NLP applications automatic text summarizing parsing sentiment analysis are part of RNN (Recurrent neural network).

Machine learning algorithms cannot deal with text direct it should be converted into numerical value one method could be representing each word into one hot vector**.** One hot vector can tell nothing about semantic of words.

However, neural networks do not produce any words. They calculate the likelihood of each word appearing next in the sequence. The concept of one-hot encoding of a word is used by neural networks. A vector of one-hot representation is a vector of one-hot representation is a vector of one-hot representation V dimensions, where V is the size of the input data's vocabulary. Each index is unique.

Represents a word in the neural system and that index is constant. The information provided One-hot vectors are used to represent input and output words**;** however, there is a distinction to be made. Internal representation and external representation only the word's index is included in the input vector.

All indices have a value of 0 except one, which has a value of 1. As a result, the output vector is a prediction vector.

Each index reflects a different type of data. For example, model trained the following sentence “I drink water “if model knows relationship between Words, the word milk close to water and far from shoes. Then system decide that “I drink shoes “is not a valid sentence another example look figure (2.4) countries and capitals there is relationship between Ankara and Turkey this means similar words cluster close to each other.

2.3 Brief Scientific Background Information About Text Generation

Summarization gained popularity in the field of research early 1950s. Luhn was first automatic text summarization system. Luhn proposed heuristic method which was counting the significant of words. for example, if sentence length is 10 words may be 4 significant words after removing stop words and steaming. Most Frequency words are not significant (and is,am,are) (Lunh, 1958).

Edmanson(Corporation et al., 1964) proposed new statistical automatic extraction of cue, key, title and location method. He emphasized that semantic and syntactic feature of the text should take account.

The earliest classification model for the automated text summarizer was introduced by Moens and Moens, (1986), using a corpus of technical papers connected with summaries written by experienced authors. The method classifies sentences with their description in text. Then get a model which predict new data and generate abstract.

Naïve Bayes classifier is supervised machine learning approach which used labeled dataset and predict new data by using corpus (Naidu et al., 2018)

Lapata and Cheng (2016) A sentence extractive model has been developed that uses a CNN word level to encrypt sentences and a sequence-to-sequence model of sentence level to forecast, which phrases include in the summary.

Cheng et al., (2020) proposed similar neural text summarization by using syntax augment which is the Sentence embeddings integrate syntactic structure information, and attention to syntactic units is paired with attention to sentences to guide summary generation.

2.4 Attention Mechanism

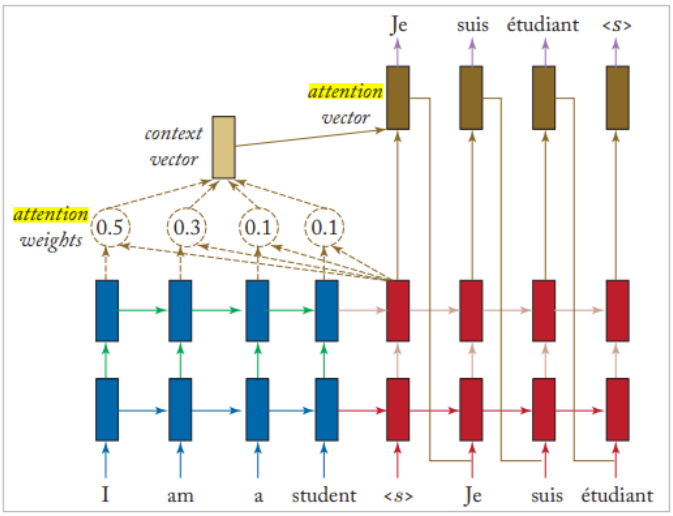

Figure 2.4: Long short-term memory (LSTM) with attention model

The above diagram (figure 2.4) shows attention LSTM cell with attention layer. Attention layer decides which part of the text should be memorized in order to generate summary.(Rush et al., 2015)

3. METHODOLOGY

3.1 Model Diagram

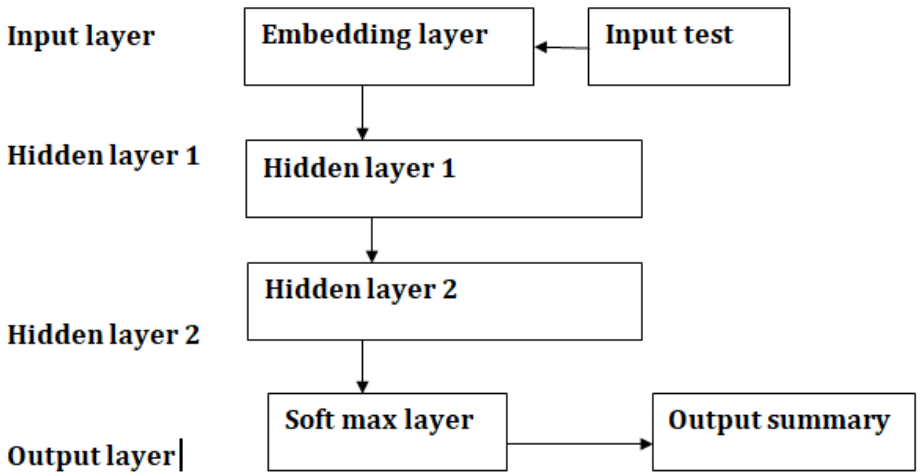

Figure 3.1 Model diagram

Previous chapter we discussed state of art of text summarization, learned converting words into vectors using word embedding, then first step being taken to preprocess text then feed network and max layer calculate which words or probability of words which include summary and final summary generated

3.2 Long Short Memory (LSTM)

In 1997, the Long Short-Term Memory (LSTM) cell was suggested, except that it will do even better; training will converge more easily and detect long-term data dependencies. Hochreiter and Urgen Schmidhuber (1997) the diagram below show LSTM architecture, it has multiple gates: input, forget, new memory and output gate.

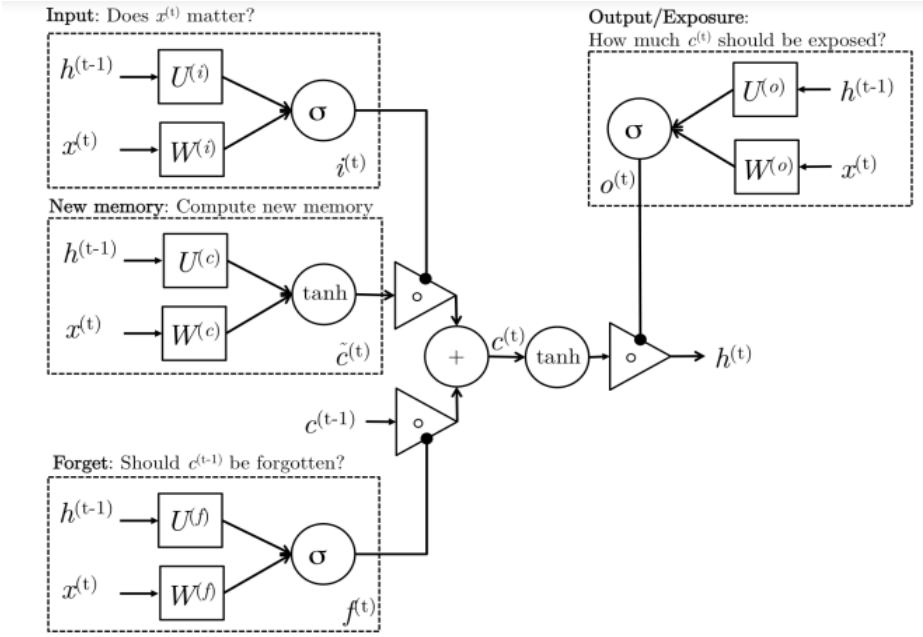

Figure3.1: Long short time memory (LSTM)

3.3 Algorithm Steps

1. Data set read
2. Preprocess data
3. Check if data cleaned if yes then go next step else back preprocess data
4. Add word embedding layer
5. Add special tokes like eos ukn
6. Encoder decoder with Lstm.
7. Sequence generation
8. Train model
9. Summary generation

3.3.1 Data set reading

This step including reading dataset using panda frame work.

3.3.2 Preprocessing data

Chapter one discussed data preprocessing convert crude data into more clean data in order to be suitable for further analysis, Natural language tool kit (NLTK) used to tokenize sentence into words, Porter Stemmer is used to obtaining stems of the words in the text. After tokenization, it is divided into 16-word-length sequences. In order to feed numerical values to LSTM word embedding is utilized.

3.3.3 Check if data cleaned if yes then go next step else back preprocess data

This step created function to verify where text preprocessed if not, we preprocess and remove stop words.

3.4.4 Encoder decoder with LSTM.

The encoder reads the input sequence and generates a fixed-length vector. Following that, the decoder utilizes this fixed-length vector as an input to the first of its LSTM units, together with the output hidden and cell states. The decoder generates a fixed-length vector that we will use to determine the target label.

3.3.5 Train model

The gates use sigmoid activation functions which are quite similar to tanh hyperbolic activation. They squish the values between 0 and 1. 0 blocks everything while 1 makes everything pass forward.

3.3.6 Summary generation

converting words into vectors using word embedding, then first step being taken to preprocess text then feed network and max layer calculate which words or probability of words which include summary and final summary generated.

3.3.7 Long short-term memory (LSTM)

LSTM units are made up of three gates; Input Gate, Forget gate and the Output Gate. The gates decide which information is relevant to and has to pass further or can be forgotten during training. The gates use sigmoid activation functions which are quite similar to tanh hyperbolic activation. They squish the values between 0 and 1. 0 blocks everything while 1 makes everything pass forward. These gates also help in tackling the problem of vanishing or exploding gradients through a gating mechanism, In a forget gate, the gate decides whether information should be discarded or maintained, hence a sigmoid activation is utilized, with an output closer to 0 indicating that it should be discarded and an output closer to 1 indicating that it must be kept the same information is passed through tanh function to adjust the network. Then the two outputs multiplied.

$$i^{(t)} = \sigma\left(W^{(i)}x^{(t)} + u^{(i)}h^{(t-1)}\right) \qquad (input\ gare) \qquad (3.1)$$

$$f^{(t)} = \sigma\left(W^{(f)}x^{(t)} + u^{(f)}h^{(t-1)}\right) \qquad (output\ gate) \qquad (3.2)$$

$$o^{(t)} = \sigma\left(W^{(o)}x^{(t)} + u^{(0)}h^{(t-1)}\right) \qquad (Forget\ gate) \qquad (3.3)$$

$$\tau^{(t)} = tahn\left(W^{(c)}x^{(t)} + u^{(0)}h^{(t-1)}\right) \qquad (new\ Memory\ Cell) \qquad (3.4)$$

$$i^{(t)} = f^{(t)}o\tau^{(t-1)} + i^{(t)}\ o\tau^{(t)} \qquad (Final\ Memory\ cell) \qquad (3.5)$$

$$i^{(t)} = o^{(t)}\ o\ \tanh(c^{(t)}\ ) \qquad (3.6)$$

3.4 Sequence to Sequence

3.4.1. Language model

Probability model calculates probability of sentence to predict next words

P (A | B) = P (A, B) / P (B)

Rewriting P (A,B) = P(A | B) P(B)

P(A,B,C,D) = P(A)P(B|A)P(C|A,B)P(D|A,B,C)

- The Chain Rule in General

$P(x_1,x_2,x_3,\ldots,x_n) = P(x_1)P(x_2|x_1)P(x_3|x_1,x_2)\ldots P(x_n|x_1,\ldots,x_{n-1})$

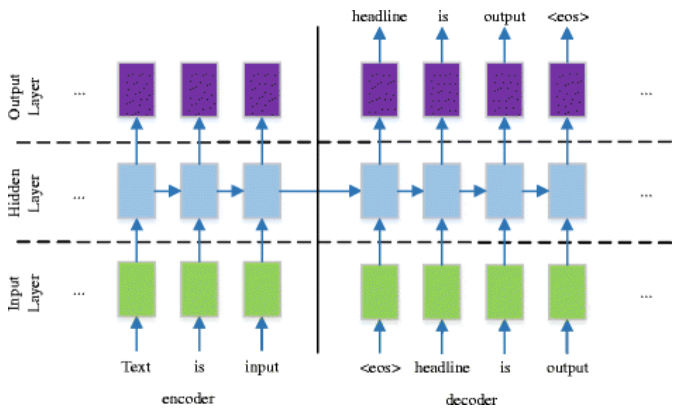

Figure 3.3 Sequence to sequence model architecture

An encoder and a decoder are the two main components of the model. Recurrent Neural Networks (RNNs) are used in both the encoder and decoder. In addition to the hidden and cell states from the LSTM unit, the encoder reads the input sequence and generates a fixed-length vector. Following that, the decoder utilizes this fixed-length vector as an input to the first of its LSTM units, together with the output hidden and cell states. The decoder generates a fixed-length vector that we will use to determine the target label. We'll forecast one character at a time, making it simple to compare sequences of various lengths from one observation to the next.

3.5 Dataset and Training

Dataset obtained from Kaggle news summary public data set, which is comma separated values data type which contain two columns [news, headline].

The dataset comprises articles with news headlines, it contains 98403 records scraped the news articles from Hindu, Indian times and Guardian. Time period ranges from February to august 2017. (Vonteru, 2019)

We use Tensor flow to build our model and train on google-colab (see Figure7), which is python notebook hosted in cloud run on web browser open-source program used training natural language

3.6 Using Google Colaboratory

Carneiro et al., (2018) Google colaboratory refered to “google colab” is research prototyping machine learning model massive computational power GPUs (graphic process unite) and CPUs (central process unite). Google colaboratory is cloud services-based working on browser, it gives you free 12GP RAM of tesla K80 Accelerator which boos about 5-10x normal CPU (see figure 3.4)

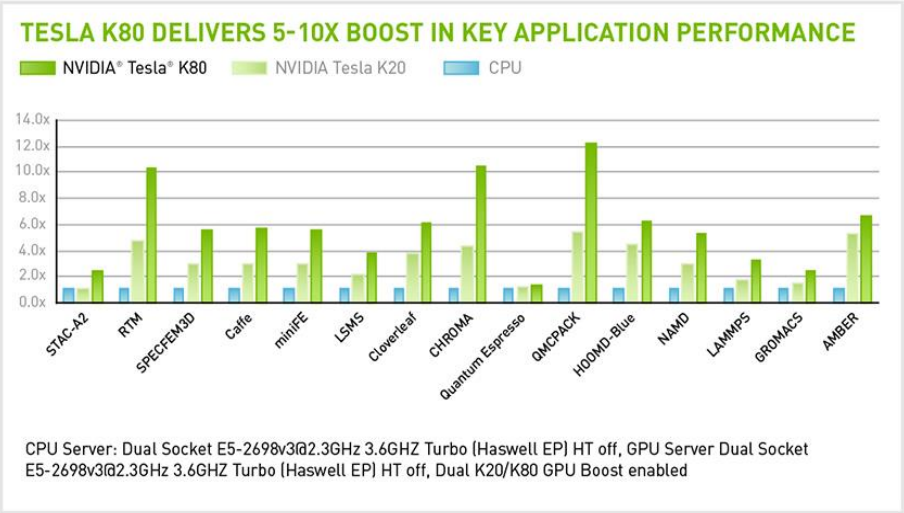

Figure 3.4 TESLA K80 vs CPU

3.7 Lunching Google Colab

The following steps shows working on google colab

1. Go to https://colab.research.google.com/ then loging your gmail account id to accecss google colab

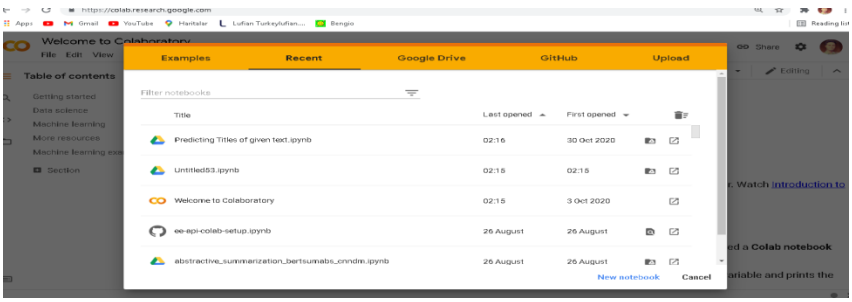

Figure3.5. Google colab home page

2. Open python 3 note book

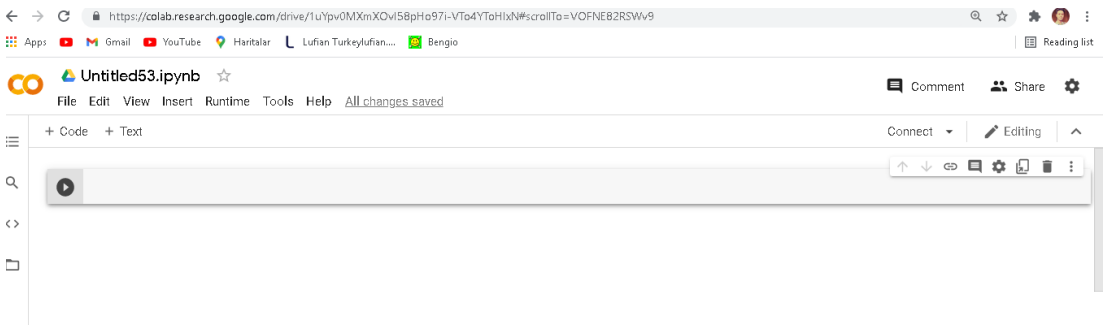

Figure 3.6 Python notebook

3.8 Benefits of Google Colaboratory

a. Zero configuration require
b. Free access Gpu
c. Easy sharing

4. EVALUATION RESULTS AND DISCUSSIONS

4.1 Rouge

Recall-Oriented Understanding for Gisting Evaluation (ROUGE) (C. Y. Lin, 2004)is metrics which evaluated automatic generated summary. It compares generated titles or summary to reference summary generated by human. It counts overlaps between summary and its reference.

Rouge 1: counts unigram overlap of word to reference
Rouge 2: counts bigram overlap of words to reference summary.
Rouge L: longest common subsequence longest counts co-occurrence to reference summary Rouge calculated the following formula

$$ROUGE - N(S|Re\ f\ s) = \frac{\sum R \in Refs \sum g_n \in RC(g_n,S,R)}{\sum R \in Refs \sum g_n \in RC(g_n,R)} \quad (4.1)$$

Rouge-n is n-gram recall between generated summary S and References (Refs). Where g_n is an n-gram,C(g_n,S,R)is the number of times that g_n co-occurs in S and reference R, and C(g_n,R)is the number of times g_n occurs in reference (Zhong et al., 2020) Other popular evaluation methods include Bilingual Evaluation Understudy (BLUE) scoring method proposed by Kishore papini, et al (2002) his approach works counting matching n-grams in the candidate sentence to reference summary.

Table 4.1: Average rouge scores

MODEL	Rouge type	Recall	Average_ Precision	Average_F1:
LSTM	Rouge 1	0,69924	:0,69924	0,69905
	Rouge 2	0,69874	0,69895,	:0,69884
	Rouge L	0,69829	0,69829	:0,69829
Seq 2 seq	Rouge 1	0.51821	0.5555	0.53623
	Rouge 2	0.19518	0.2125	0.20347
	Rouge-L	0.5182	0.55555	0.5362

These results shows that LST model scores higher than sequence to sequence model. And generated correctly most of trained examples.

4.2 Results and Discussions

The diagram shows system generated outputs which we first feed text document in neural network, we first preprocess the text

News	Human generated Titles	System predicted title
1. Brazilian mother narrates football matches to blind son from stands."Silvia Grecco, a 56-year-old Brazilian mother narrates her local football team Palmeiras' matches live to her 12-year-old blind and autistic son Nickollas from the stands. ""I describe details: this player is wearing short sleeves...colour of...football boots, hair colour...Everything I see and feel, I tell him, even when I need to curse the referee!"" Grecco said after a recent match."	Brazilian mother narrates football matches to blind son from stands	Brazilian mother narrates football matches
2. Dubai International Airport has said that it retained its position as the world's busiest airport for international travel for the fifth consecutive year in 2018. Passenger traffic at the airport rose to over 8.9 crore in 2018. India remained the biggest source of traffic, accounting for more than 1.2 crore passengers.	Dubai Airport ranked world's busiest int'l airport for 5th year	Dubai Airport ranked world's busiest int'l airport for 5th year
3. Cristiano Ronaldo converted a penalty to seal a 2-1 win for Juventus against Lazio in Rome on Sunday to pull 11 points clear in the Serie A. Ronaldo, whose penalty was saved last week against Chievo, scored in his eighth consecutive away match to bring his league tally to 15 goals. Ronaldo is currently the league's second-highest goalscorer.	Ronaldo converts penalty to help Juventus beat Lazio 2-1	Ronaldo converts penalty to help juventus.
4. Ex-Defence Minister George Fernandes, who passed away on Tuesday, forced Coca-Cola to discontinue its Indian operations in 1977 when he was the Industries Minister. Fernandes asked Coca?\x9b???\x90Cola to transfer 60% of ownership in Indian operations and its formula to a local company. The US Company said it was agreeable to transferring a majority of the shares but not the formula.	Ex-Defence Minister Fernandes once forced Coca-Cola out of India	Minister fernandes forced Coca-cola out of India.

Figure 4.1: System predicted titles

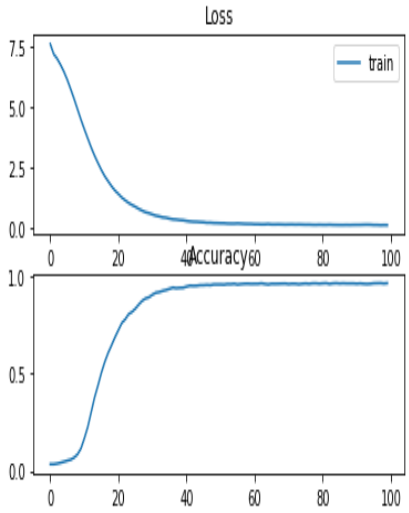

Figure 4.2: Accuracy and los during training

5. CONCLUSION AND IMPLICATIONS

This study emphasized neural network document summarization model, which predict title of given text. We feed neural network encoder decoder to generate text.

Decoder generates a word as input when generating the next word until it has reached actual headline. All weights are sum and squished threshold.

As the availability of tremendous text date on online increase using automatic summarizers saves time and effort to users. Getting salient and syntactic headlines is quite challenge, although current researches solved many of them, deep learning need more text of data to train and predict good result.

REFERENCES

Allahyari, M., Pouriyeh, S., Assefi, M., Safaei, S., D., E., B., J., Kochut, K., 2017. Text Summarization Techniques: A Brief Survey. International Journal of Advanced Computer Science and Applications, 8(10).

Boehmke, B., Greenwell, B., 2019. Hands-On Machine Learning with R. In Hands-On Machine Learning with R. OReilly., Chapman and Hall/CRC.

Carneiro, T., Da Nobrega, R. V. M., Nepomuceno, T., Bian, G. Bin, De Albuquerque, V. H. C., Filho, P. P. R., 2018. Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications. IEEE Access, 6, 61677–61685.

Cheng, J., Zhang, F., Guo, X., 2020. A Syntax-Augmented and Headline-Aware Neural Text Summarization Method. IEEE Access, 8, 218360–218371.

Chopra, D., Mathur, I., Joshi, N., 2016. Mastering Natural Language Processing with Python : Maximize Your NLP capabilities While Creating Amazing NLP projects in Python. PACKT.

Corporation, T. B., Park, C., Orleans, N., Command, E., Food, Q., Press, P. U., 1964. Problems in Automatic Abstracting Document] Automatic ~ Extracting I Extract I Generative ~ Grammar. 259–263.

Hermawan, R., 2011. Natural Language Processing with Python. in Indonesian Journal of Applied Linguistics, 1(1).

Hochreiter, S., Urgen Schmidhuber, J., 1997. Long Shortterm Memory. Neural Computation, 9(8), 17351780.

Google, 2021. Word Embedding's. Date Accessed : 29.08.2021, https://developers.google.com/machine-learning/guides/text-classification/images/WordEmbeddings.png

Jain, A. K., Mao, J., Mohiuddin, K. M., 1996. Artificial Neural Networks: A Tutorial. Computer, 29(3), 31–44.

Kayli, l., 2019. Learning Paradigms of Neural Networks. Date Accesses: 24.11.2021. https://medium.com/swlh/learning-paradigms-in-neural-networks-30854975aa8d.

Lin, C. Y., 2004. Rouge: A Package for Automatic Evaluation of Summaries. Proceedings of the Workshop on Text Summarization Branches out (WAS 2004), 1, 25–26.

Lin, Y., Wang, X., Zhou, A., 2016. Opinion Spam Detection. Opinion Analysis for Online Reviews, May, 79–94.

Liu, Y., 2019. Python Machine Learning by Example-Second Edition. Date Accessed: 01.09.2021, https://learning.oreilly.com/library/view/python-machine-learning/9781789616729/

Lunh, H. P., 1958. The Automatic Creation of Literature Abstracts. IBM Journal of Research Development, 2(2), 159–165.

Mehryar Mohri Afshin, 2018. Foundations of Machine Learning. MIT Press.

Moens, B., Moens, C., 1986. Intra-articular Injection of Phenylbutazone in Gonarthrosis. Annals of the Rheumatic Diseases, 45(9), 788.

Naidu, R., Bharti, S. K., Babu, K. S., Mohapatra, R. K., 2018. Text Summarization with Automatic Keyword Extraction in Telugu E-Newspapers. Smart Innovation, Systems and Technologies, 77, 555–564.

Rush, A. M., Chopra, S., Weston, J., 2015. A Neural Attention Model for Sentence Summarization. Conference Proceedings - EMNLP 2015: Conference on Empirical Methods in Natural Language Processing, 379–389.

Russell, R., 2018. Machine Learning Step-By-Step Guide to Implement Machine Learning Algorithms with Python. 106. Date Accessed : 01.09.2021, http://booksdescr.org/item/index.php?md5=D161EE832B8007A058CD006DD67E388E

Sam, F., 2018. How NLP, ML and Deep Learning Can Transform Your CX Strategy. Date Accessed: 22.11.2021. https://www.business2community.com/customer-experience/how-nlp-ml-and-deep-learning-can-transform-your-cx-strategy-02137651

Sara. B., 2021. Machine Learning Explained, Date Accessed: 04.10.2021. https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained

See, A., Liu, P. J., Manning, C. D., 2017. Get to the Point: Summarization with Pointer-Generator Networks. ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers), 1, 1073–1083.

Trilla, A., 2009. Natural Language Processing Techniques in Text-To-Speech Synthesis and Automatic Speech Recognition, 1–5. Date Accessed : 01.09.2021, http://atrilla.net/data/files/micnlp09.pdf

Vonteru, 2019 News Summary Date Accessed : 21.06.2021 https://www.kaggle.com/sunnysai12345/news-summary

Zhong, M., Liu, P., Chen, Y., Wang, D., Qiu, X., Huang, X., 2020. Extractive Summarization As Text Matching. Date Accessed : 01.09.2021, https://doi.org/10.18653/v1/2020.acl-main.552

APPENDIX

```
import numpy as np
import pandas as pd
import tensorflow  as tn
#!pip install rouge
#from rouge import Rouge
import string
import nltk

from google.colab import files
uploaded = files.upload()

%matplotlib inline
import matplotlib.pyplot as plt
plt.style.use('seaborn-whitegrid')
import numpy as np

from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Embedding, LSTM, Dense, Dropout, Bidirec
tional
from tensorflow.keras.optimizers import Adam
import tensorflow.keras.utils as ku
import numpy as np

df = pd.read_csv('News_Sumarry_More.csv', encoding='iso-8859-1')
df.head()
```

	headlines	text
0	upGrad learner switches to career in ML & Al w...	Saurav Kant, an alumnus of upGrad and IIIT-B's...
1	Delhi techie wins free food from Swiggy for on...	Kunal Shah's credit card bill payment platform...
2	New Zealand end Rohit Sharma-led India's 12-ma...	New Zealand defeated India by 8 wickets in the...
3	Aegon life iTerm insurance plan helps customer...	With Aegon Life iTerm Insurance plan, customer...
4	Have known Hirani for yrs, what if MeToo claim...	Speaking about the sexual harassment allegatio...

Data preprocessin to lower case

```
df['text_lower']=df['text'].str.lower()
df['headlines_lower_Case']=df['headlines'].str.lower()
df.head()
```

```
[ ]  df.text_lower.head()
```

```
0    saurav kant, an alumnus of upgrad and iiit-b's...
1    kunal shah's credit card bill payment platform...
2    new zealand defeated india by 8 wickets in the...
3    with aegon life iterm insurance plan, customer...
4    speaking about the sexual harassment allegatio...
Name: text_lower, dtype: object
```

```
# drop the new column created in last cell
#df.drop(["text_lower"], axis=1, inplace=True)

PUNCT_TO_REMOVE = string.punctuation
def remove_punctuation(text):
    """custom function to remove the punctuation"""
    return text.translate(str.maketrans('', '', PUNCT_TO_REMOVE))

df["text_wo_punct"] = df["text"].apply(lambda text: remove_punctuation(text))
df.head()

Remove Stop Words
from nltk.corpus import  stopwords

#defining function for tokenization
import re
def tokenization(text_wo_punct):
    tokens = re.split('W+',text_wo_punct)
    return tokens
#applying function to the column
df['msg_tokenied']= df['text_wo_punct'].apply(lambda x: tokenization(x))

STOPWORDS = set(stopwords.words('english'))
def remove_stopwords(text):
    """custom function to remove the stopwords"""
    return " ".join([word for word in str(text).split() if word not in STOPWORDS])

df["text_wo_stop"] = df["text_wo_punct"].apply(lambda text: remove_stopwords
(text))
df.head()

STOPWORDS = set(stopwords.words('english'))
def remove_stopwords(text):
    """custom function to remove the stopwords"""
    return " ".join([word for word in str(text).split() if word not in STOPWORDS])

df["text_wo_stop"] = df["text_wo_punct"].apply(lambda text: remove_stopwords
(text))
```

```
df.head()

from nltk.stem.porter import PorterStemmer

# Drop the two columns
#df.drop(["text_wo_stopfreq", "text_wo_stopfreqrare"], axis=1, inplace=True)

stemmer = PorterStemmer()
def stem_words(text):
   return " ".join([stemmer.stem(word) for word in text.split()])

df["text_stemmed"] = df["text_wo_stop"].apply(lambda text: stem_words(text))
df.head()

from nltk.stem.porter import PorterStemmer

#Drop the two columns
df.drop(["headlines_lower_Case", "headlines_stemmed"], axis=1, inplace=True)

stemmer = PorterStemmer()
def stem_words(headlines):
   return " ".join([stemmer.stem(word) for word in headlines.split()])

df["headlines_stemmed"] = df["headlines_lower_Case"].apply(lambda headlines:
 stem_words(headlines))
df.head()

for l in headlines[:5000]:
    token = tokenizer.texts_to_sequences([l])[0]
#     print(token)
    for i in range(1,len(token)):
      ngrams_seq=token[:i+1]
      sequences.append(ngrams_seq)
for i in sequences:
   k=len(i)
   if k>maxl:
      maxl=k
data= pad_sequences(sequences, maxlen=maxl)
data
```

```
data= pad_sequences(sequences, maxlen=maxl)
data
```

```
array([[   0,    0,    0, ...,    0,  758,  759],
       [   0,    0,    0, ...,  758,  759,  760],
       [   0,    0,    0, ...,  759,  760,    1],
       ...,
       [   0,    0,    0, ..., 2479,    3, 2480],
       [   0,    0,    0, ...,    3, 2480,  737],
       [   0,    0,    0, ..., 2480,  737,   87]], dtype=int32)
```

```
predictors=data[:,:-1]
predictors

model = Sequential()
model.add(Embedding(input_dim=total_words,output_dim=80,input_length=15
))#input length is 15 not 16 as we have taken the last column for labels for 16-
1=15
model.add(Dropout(0.2))
model.add(Bidirectional(LSTM(units=150,return_sequences=False)))#if return
sequences is false,then it will return a 2-D array,if true then it will return a 3-
D array..
model.add(Dense(total_words,activation='softmax'))

model.summary()

model.compile(optimizer='adam',loss='categorical_crossentropy',metrics=['acc
uracy'])

history = model.fit(predictors, labels, epochs=100, verbose=1)

accuracy = history.history['accuracy']

epochs = range(len(accuracy))

plt.plot(epochs, accuracy, 'b', label='Training accuracy')
plt.title('Training accuracy')
```

BIBLIOGRAPHY

Name Surname : Mohamed Barre OMER

Education background

High school : Hamdan sencondary school , 2008

Degree : Hargeisa University , Informatoin Technology

Masters Degree : İstanbul Commerce University , Science Institute, Computer Engineering

Publications

Omer, M. B., Kasapbaşı, M. C., 2021. Predicting Title of Given Text Using Deep Learning.4th International Conference on Data Science and Applications (ICONDATA21), 77.