National College of Ireland

Predicting the Ranking of Web Page on SERP by Applying Machine Learning Techniques

MSc Research Project
Data Analytics

Shubham Garg
Student ID: x19205295

School of Computing
National College of Ireland

Supervisor: Martin Alain

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Shubham Garg
Student ID:	x19205295
Programme:	Data Analytics
Year:	2021
Module:	MSc Research Project
Supervisor:	Martin Alain
Submission Due Date:	16/12/2021
Project Title:	Predicting the Ranking of Web Page on SERP by Applying Machine Learning Techniques
Word Count:	6717
Page Count:	21

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Shubham Garg
Date:	31st January 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Predicting the Ranking of Web Page on SERP by Applying Machine Learning Techniques

Shubham Garg
x19205295

Abstract

Analyzing the rank of web pages related to the query is always been a problem and many organization work for it to improve the ranking in search engine result page. It depends on more than 200's of factors but no one can justify its original algorithm through which it can be enhanced quickly with correct input. It changes regularly based on the meta data behind the web page but also it depends upon the no. of visitors on web page, total no. of time they spend on it etc. this project aim to predict the ranking on the basis of meta data , title and snippet of web page. The proposed architecture is Recurrent Neural Network(RNN) by using the Long Short Term Memory(LSTM). and after applying the proposed model it achieve the accuracy of 64%. with the loss of 0.93. Their are several other models have been compiled but no can get the accuracy of more than 64%. The model has been compared in the reference of Dense Layer and weights. The batch sizes and Epochs have been changed to achieve more accuracy. Along with this the EDA is also performed where it shows that how the ranking of web page impact on traffic visiting the websites.

1 Introduction

As the world is moving forward towards digitalization, the search engine is the most common thing in a day-to-day life of every person on the earth. For being more reliable and feasible many engineers are working and have gained so much efficiency that a user can find his related work on the search engine result page. Earlier we have to recall the Ip address linked to the website bust is difficult to remember every so the domain name was started as the name are easy to recall and remember every website has their unique web address and name as well. But in the increasing of the data remembering of the website name is difficult and impossible to find the new websites so the search engine introduced in year 1990 and month September named Archie. It went well with search quires as it gives mix of several things related to the query such as images, web links, videos, etc. it gives the real time information that mean it is not maintained by the human efforts on the other hand it maintains and update the data on the basis pre saved algorithms which works with the crawler and update the search engine result page accordingly.Drivas et al. (2017) But there are some drawbacks that which web link shows first based on the searches because most upfront web link will be opened more as compared to second and so on so page ranking and authentication is necessary so the web-crawling and indexing is necessary.

Sagot et al. (2014)The indexing is necessary which gives the ranking to the page depends on many factors such as title of the website, Hypertext markup language (HTML) name assigned to the web pages and their tags, the meta data of the web page, hash tags behind the title of the pages and the content. Every web developer and content manager keep this in the mind to give related and correct information about that web page content so that the user get the desired landing page.Khorsheed et al. (2015) After providing the information the crawler works starts, the most common crawler is spider, the crawler will scrap the data from the web pages provided by the owner of the website the it may or may not retrieve all the data based on the type of the website as it is not trusty website or a new website it may be possible to fetch only partial data because there may be chance of scams although it is a free service so many people try phishing on the user data and can get their personal details. So, the crawler work is to authenticate the websites and on the basis of that data it provides the ranking and indexing to a particular website based on the field related query. Every people and company creating their own websites and there are several tools on online where we can create new websites according to our requirements easily and the basic websites are available instantly there is no need of coding background and are very cheap although finding the domain has become easier now a days through which you can create and publish your websites in a few hours. But the user who is browsing his related query and desired websites are hard to find, so GOOGLE which the biggest search engine is using by everybody in the world performing many algorithms behind the screen and making easier for us to search. They have also introduced filters in the general searches so that only related to that field will be shown for instance if I want to search for images so, I can select the searches, or want to book a flight will direct me to only flight pages similarly for shopping it only shows me the websites related to shopping only etc. keeping this in the mind we have the data of search engine result page which is filtered by only related to flights and every first Ten ranked pages are scraped for the flights to different destinations like Hong Kong, Bangkok, London, Singapore etc. It is an open-source data which available in Kaggle it is also an open-source directory of datasets where we can find the different types of datasets as per our requirements. The dataset consists of 4000 rows of 400 different destination search queries and collected every month for a period of two years which makes it the dataset of 48000 rows. The goal of our project is to determine that what are the basic fundamentals of the website which makes the ranking and indexing on the result page of the search engine. And what are their impact with combined and individually on the indexing of the website.
we have first analyse the dataset and getting the inference from it to perform the electronic design automation and following this we have applied the Long Short-Term Memory (LSTM) which is an architecture of RNN (Recurrent Neural Network). In this model we have performed several tasks in which we have changed the number of dense layer and their sizes to get the best result out of them.

1.1 Research Question

To predict the web page ranking based on the textual data and To analyze that how the traffic is high on the top ranked web-pages as compared to the lower ranked web-pages.

Table 1: Research Pattern

Objective	Description
1	The reviews existing researches
2	Methodology (EDA and Pre processing)
3	Implementation
4	Evaluation of models and results
5	summarizing the results and future wok

2 Related Work

In this part of the paper, it will give the reviews of the paper used for analysing and predicting the rank of webpages by different researchers. Apart from this this part will also tell about the implementation and methodologies can be improvised.
Salminen et al. (2019)Researched on prediction of ranking on the gift industry webpages on the factor of search engine optimization. They have used two machine learning algorithms that are XGBoost and LightBGM to predict the ranking. They performed on the ecommerce website data so they have the data of keywords integrated with the webpages. They work on the Learning to rank algorithm where they try to predict the nearly absolute rank based on the NDGC score. As long as the NDGC score is high for as particular keyword will be most likely to get the highest rank but the limitation of this research is that it is biased towards one company although it is in the Finnish language so they have to convert it into English language which may impact on loss of data. Another limitation is that the factors are very limited and in real world search engine where many companies are in the queue of competition have many other factors like no. of views, meta data of the webpage etc. But the good part is they have analysed the given dataset thoroughly and pick out the most valuable keywords which is useful for single organization in itself. Zhang and Dimitroff (2005)researched on the metadata and title effectiveness by applying the hypotheses. They have applied three different techniques of statistics which are one way ANNOVA, independent sample T-test and two way ANNOVA. Through these tests they worked on H1 text, H2 text, H3 text, H4 text. These H are the headings in the webpage as the number increases the heading decreases and becomes the sub section of larger H. they have performed the experiments on 46 different webpages. Their researchers monitor and recode the data weekly on 19 different search engines. They performed it till 21 weeks. And by performing their methods they get the result that elements of metadata give the better ranking and performance compared to who don't have metadata. Arora and Bhalla (2014)In addition to which we have applied the neural networks on meta data, snippet and on title as well.Sen (2005)researched and provide the same result for the metadata. But he also proposed the theory that most of the organization used the paid listing on the search engine rather than working on the optimization on the search engine by the method of search engine marketing.
Evans (2007)researched on the optimization of the webpages they have taken 50 optimized web pages and 50 non optimized for the purpose of comparison. Their purpose of working is to provide the enhancement of the new or old website or an e-commerce organization to get the wide area of networking and increasing the visitors. In their research they said that over 200 factors which decide the ranking on the google search engine result page. The issues they were faced that combining and analysing all the factors of ignored query and query independent is not possible on the resources available currently

Figure 1: Khorsheed et al. (2015) proposed model

but in the future, it may be possible to do the learning of ranking on these factors. But Cunningham (1998)they conclude that score of page rank, links in links, listing in different directories and the last is how old is the domain of the website.similarly Wang et al. (2011) researched on the data collected from different 116 websites which were done by the tool of third party. They focused on optimization of structure, link, content, keywords. In result they shows that smaller the size of webpage, precise and less title characters and 2 to 8% density of keywords gives the best result in getting the good ranking of a webpage.Mandl (2006)A method for implementing a quality-based Web crawler is presented in this thesis. The most effective methods for determining the quality of Web sites have been recognized as ml algorithms. Based on human assessments, a quality model is created and implemented into a meta web browser. Search engine were created on java and from the directories of internet quality web pages were retrieved and then compared with web pages randomly. The WebTango gives the features more than 150 and tried to get the corelation in the websites which is highly rated. But it is limited because of the assumption of theory on which the research was working.
Chauhan et al. (2015)]This study contains an extensive literature review on machine learning-based html pages ranking algorithms. The architecture of the web was mostly exposed via page ranking. Automation also aids us in comprehending the more sophisticated aspects of page setting objectives in prominent search engines such as Google, Yahoo, AltaVista, dog pile, and others. This author used Random walk where the web crawler moves from one node to another neighbour node randomly. It helps to find the relevant page for the query because the neighbouring nodes are only related to the query searched. They also performed clustering where they define that a particular search shows the all aspects of the results which every possible. If we search flight in the query it not only show the flights booking but also the flight mode in mobile phone as well as the pictures of flights etc. the algorithms were used by them are HITS, DIRECTHIT in these HITS give the best related query result where as DIRECTHIT works on the click on the web pages. In conclusion they said that if the page is linked by any medium to the high ranking page then the probability of their good ranking are high.Lee et al. (2016)a case study to examine the SEO tactics used on LG Science Land material, and discovered the following variables to be important: a streamlined Http structure, an administrative redirect in the event of a webpage deletion, a Xhtml sitemap to aid web search indexing, informative titles as well as meta-tags, normative URLs, and the disposal of outdated hyperlinks and material. Khekare and Verma (2020)The Internet of Everything (IoE) makes

it possible to link everything to the web. This article proposes a one-of-a-kind technique for identifying critical keys. Speech instructions can rapidly and efficiently obtain needed data from such keys. The entire procedure is updated and k-means clustering is applied. The data is obtained by the voice command utilizing nlp. The suggested system has produced superior results, and the process of finding has grown easier and more efficient as a consequence of simulation findings. The have used the similar steps as of this research used which is tokenization, removal of stop words, TF-idf vectorizer and then applying the method of clustering and T-SNE. To find the distance between two node they have used the formula tahts is DunnIndex = minimum(inter cluster distance) / maximum (intra cluster distance). The researchers have provided very efficient and better solution but it only find the keywords for providing the organisation to enhance the rank of the web page. Kumar et al. (2016) proposed the strategies used to improve the ranking of a web page might be characterized as web spam. Content spam, link spam, and cloaking spam are the three most common forms. They offer a classifier called the Dual-Margin Multi-Class Hypersphere Support Vector Machine (DMMH- SVM). They also developed new cloaking-based spam features that aid our classification algorithm in achieving high accuracy and recall rates, lowering false positive rates. DMMH-SVM surpasses current algorithms with unique cloaking properties, according to the results of this study.Eirinaki et al. (2005) Non - cyclic path and valued allocated to any html pages path prediction is done after the forecast of html pages rating grid is contrasted with the frequency using a Markov model for trying to match the outcome.Singh and Gupta (2013)The Internet of Everything (IoE) makes it possible to link everything to the web. This article proposes a one-of-a-kind technique for identifying critical keys. Speech instructions can rapidly and efficiently obtain needed data from such keys. The entire procedure is updated and k-means clustering is applied. The data is obtained by the voice activation utilizing nlp. The suggested system has produced superior results, as well as the process of finding has grown easier and more efficient as a consequence of simulation findings.

Hu et al. (2018) proposed the Reinforcement learning methodology to predict the rank in the search engine used by the organisation of e-commerce. They first performed the SSMDP(search session Markov decision process) to know the prroblem in multi steps of the process of ranking. and then they try to prove theoretically of maximizing the rewards accumulative. and in the end gradient algorithm is proposed. in their result they shows that the accuracy achieved by their proposed systems are much better than the LTR methods which is more likely to be 40% in the amount of transaction of siulation and 30% in the real world applications. For the future work they said that the challange will be more likely to be dynamic environment.

The previous researches have done either on the single organisation or the weight-age of keywords by applying the reinforcement learning and XGboost model respectively. so this paper proposed a methodology that will predict the ranking on the basis of keywords in the meta data , title and snippet. By applying the deep learning methods that is Recurrent Neural Network by adding layers of LSTM , dense layer, embedded layer, Spatial Dropout1D. so this research will take one step forward towards the prediction of ranking based on the textual data stored in back-end code of the web page.

3 Methodology

To perform this task and to achieve our goal we have performed several steps these steps are the parts of the methodology Knowledge Discovery (KDD). Apart from this there is another method which is also performed by many researchers is Cross Industry Standard Processes for Data Mining (CRISP-DM). But from these two distinct methodologies we only perform KDD.

3.1 KDD

From the name itself we can understand that it is proceed with several steps which are as follows:

3.1.1 Insights of business

In the very first step we will be looking for the output that what we will be getting from this experiment and how it will be useful for future process and can ease the everyday life as because our project is on search engine optimization so we will be concluding the result that how are search engine provides ranking on the basis on meta data, title and snippet of the website. Through this we will be seeing that by applying the model how much accuracy we can get to improvising the rank of website as it may be overlapping the goals also may be influencing to the firm or organization which should managed carefully. To conclude it we can say that this process will highlight some main factors and features have more impact on the result of the project.

3.1.2 Insights of data

Before getting insights, we must have the data, the data may be provided by the organization or may be mined through website as it is an individual project so, I have downloaded the data from Kaggle. It is an open-source directory from where we can download or upload the data also, we can upload our notebook in the reference of the data. So, the we have chosen a dataset which have 43 columns and 4000 rows we also have used another dataset which is all similar to this dataset but has 26 column and 4000 rows but also, we have this data which is scrapped twice in every month regularly for a period of one year and four months start from 16th of December 2018 to 1st of April 2020.
In the first dataset we have 43 columns which are mostly strings, out of which it includes title of websites, snippets, metadata, metadata with heading one, two and three, Url of the website, url of he landing page to the website, ranking of the websites which is changed based of the user interactions and how long they spend the time on the webpage also it is our target variable, search time, total no. of results related to the search query, load time for search engine result page, the size of scraped data, date and time of the scraped data. As this data is taken once so the date and time is same that is 11th of march 2020 at 17:56:16 on the other hand it is all same but many of the columns are not included but we have the most important column which are title and snippet which will be used for modelling process to evaluate the ranking apart from these we have much positive point which have huge amount of data as compared to the previous one which is 24 times. This data will be helpful for analysing and performing the Exploratory Data Analysis (EDA) which gives us the inference of the data that what and how the data is

flowing or in other words we can say that through the graph we can easily understood the data and its reference to the rank. The EDA processing and result are described below:

3.1.3 EDA

In the process of EDA our very first step is to check what is the dataset in Figure 2 and which type of data types dataset Figure 3a will give the information about it:

	searchTerms	rank	title	snippet	displayLink	link	queryTime	totalResults	cacheId	formattedUrl	htmlFormattedUrl	htmlSnippet	htmlTitle	
0	flights to hong kong	1	Cheap Flights to Hong Kong (HKG) from $397 - K...	Find flights to Hong Kong on XiamenAir, Hong K...	www.kayak.com	https://www.kayak.com/flight-routes/United-Sta...	2018-12-16 11:26:30.485612+00:00	106000000	W42baDpas_gJ	https://www.kayak.com/flight-routes/United.../...	https://www.kayak.com/flight-routes/Uni...	Find flights to Hong Kong on XiamenAir,...	Cheap Flights to Hong Kong (HKG) from $...	customsea
1	flights to hong kong	2	$480 Flights to Hong Kong, China (HKG) - TripA...	Cheap Flights to Hong Kong: Enter your dates o...	www.tripadvisor.com	https://www.tripadvisor.com/Flights-g294217-Ho...	2018-12-16 11:26:30.485612+00:00	106000000	5jsjhmxgxvkJ	https://www.tripadvisor.com/Flights-g294217-Ho...	https://www.tripadvisor.com/Flights-g29...	Cheap Flights to Hong Kong: Enter your ...	$480 Flights to Hong Kong, China (HKG) ...	customsea
2	flights to hong kong	3	Cheap Flights to Hong Kong International from ...	Search cheap flights using Skyscanner's free f...	www.skyscanner.com	https://www.skyscanner.com/flights-to/hkg/chea...	2018-12-16 11:26:30.485612+00:00	106000000	EI78TI4Q1qYJ	https://www.skyscanner.com/flights-to/hkg/chea...	https://www.skyscanner.com/flights-to/hk</b...	Search cheap flights using Skyscanner&#...	Cheap Flights to Hong Kong Internationa...	customsea

Figure 2: Dataset

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4000 entries, 0 to 3999
Data columns (total 26 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   searchTerms           4000 non-null   object
 1   rank                  4000 non-null   int64
 2   title                 4000 non-null   object
 3   snippet               4000 non-null   object
 4   displayLink           4000 non-null   object
 5   link                  4000 non-null   object
 6   queryTime             4000 non-null   object
 7   totalResults          4000 non-null   int64
 8   cacheId               3970 non-null   object
 9   formattedUrl          4000 non-null   object
 10  htmlFormattedUrl      4000 non-null   object
 11  htmlSnippet           4000 non-null   object
 12  htmlTitle             4000 non-null   object
 13  kind                  4000 non-null   object
 14  pagemap               3996 non-null   object
 15  cseName               4000 non-null   object
 16  count                 4000 non-null   int64
 17  startIndex            4000 non-null   int64
 18  inputEncoding         4000 non-null   object
 19  outputEncoding        4000 non-null   object
 20  safe                  4000 non-null   object
 21  cx                    4000 non-null   object
 22  gl                    4000 non-null   object
 23  searchTime            4000 non-null   float64
 24  formattedSearchTime   4000 non-null   float64
 25  formattedTotalResults 4000 non-null   object
dtypes: float64(2), int64(4), object(20)
memory usage: 812.6+ KB
```

(a) Datatypes of data

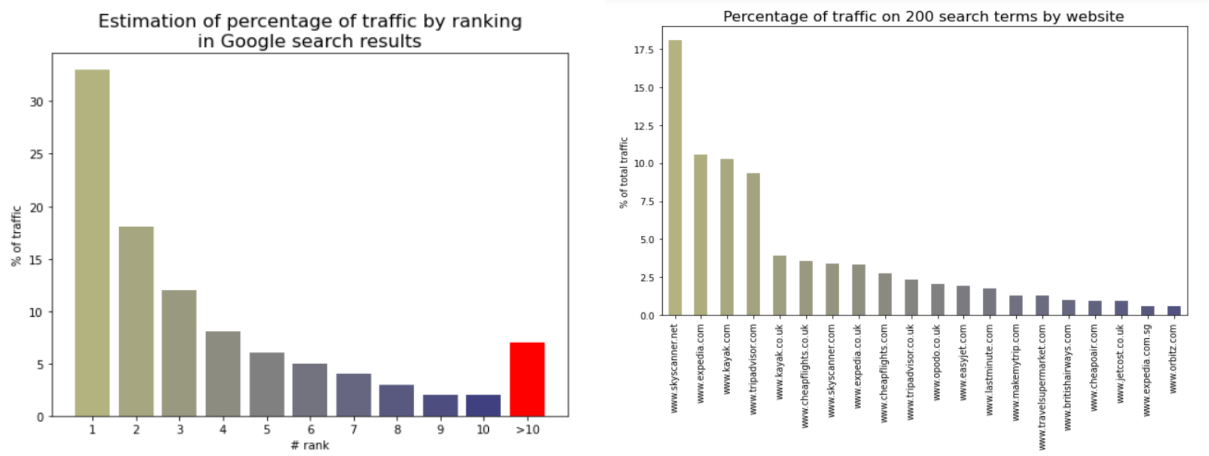
```
searchTerms             0
rank                    0
title                   0
snippet                 0
displayLink             0
link                    0
queryTime               0
totalResults            0
cacheId                30
formattedUrl            0
htmlFormattedUrl        0
htmlSnippet             0
htmlTitle               0
kind                    0
pagemap                 4
cseName                 0
count                   0
startIndex              0
inputEncoding           0
outputEncoding          0
safe                    0
cx                      0
gl                      0
searchTime              0
formattedSearchTime     0
formattedTotalResults   0
dtype: int64
```

(b) no. of null values

Figure 3: dataset information

Now we check the null or NAN values in the dataset through heat map in Figure 3b.The missing values are only on those columns which are of no use to us so we will be leaving them as they are because the page map and cache id will not be giving any inference to the rank.

Now we will be moving forward towards the most visited websites based on the rank basis in Figure 4a and the rank decreases as the traffic of visiting user is also decreases although we have data to the rank 10 and the pages more than 10 are not listed so their visitors have been combined and the result has shown above.



(a) On the Basis of Ranks

(b) On the Basis of Websites

Figure 4: Traffic percentage

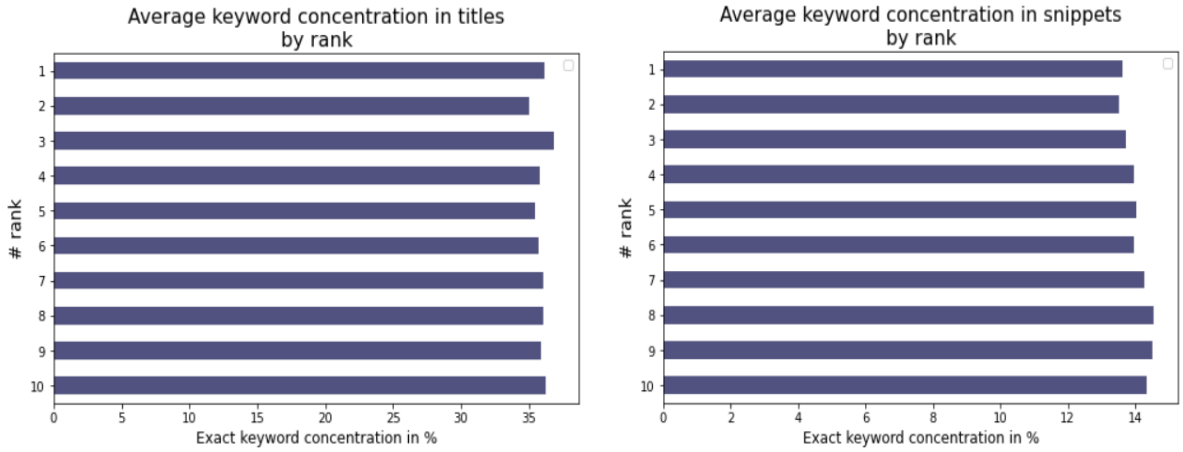
In the next step we will see the most visited websites regardless the rank by the users in Figure 4b. After seeing the above graph we can say that regardless of page ranking the skyscanner is the most visited website following with expedia , kayak , tripadvisior and so on. We can see that only four website are visited most as compared to others. The main reason behind this is they mostly shown with in top 3 ranked on result page of searche engine on the other hand the other websites are visited by the rarely that the user may be exploing the searche result of their query and also the snippets and title have been changed and forced manually by processing some black hat techniquesMalaga (2010), So they may have shown in top 10 ranked pages and visited by the user but it wont last long enough because of algorithm behind the google search engine detect it and push back with a warning and if the websites continues performing the black hat technique it may be banned by the google and wont be showing by the google search engine.
Moving further the average of concentration in titles and snippet by rank in Figure 5a and Figure 5b respectively.On looking the graph we are can say that both of the titles and snippets concentration are similar to each other but by looking closely the concentration of snippets they are increasing as the rank increased that means more precise and less concentrated keywords are more likely to achieve good rank.

After analyzing all the data in one dataset we now concatenate all the dataset of one year and 4 months and look the result that which website has gained more views as compared to others we have created it in two forms the first is all top 10 and other is top 4 in Figure 6a and Figure 6b respectively:

On analysing the above graph we can easily say that the only top 4 have gained much visitors as compared to other and out of which only sky scanner is on the top of it and has maintained its position regardless the year month and rank on the other hand the others are fluctuating by the days and months on the basis of ranks in every month.

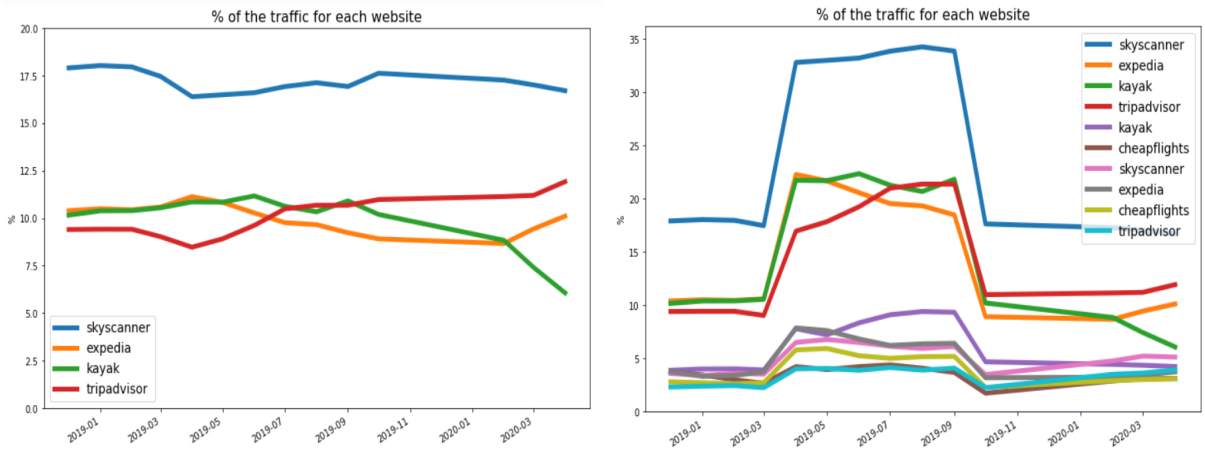
3.1.4 Pre Processing

After seeing the results of EDA we will be moved towards the pre-processing. The pre processing is the process which is used to clean the data before process with the models and predictions. Because we can't process the raw data into the models we have have to clean the impurities.



(a) Titles (b) Snippets

Figure 5: Concentration of Keywords



(a) top four Web pages (b) Top 10 Web pages

Figure 6: Traffic over the time

So in the first step we either dropping the column which is not necessary or fetch out the column into another data frame which will be needed to process the modeling. After this we will also create a new data frame of prediction variable in the data frame name 'rank_df' which consist only the column rank.

After this,the next step is of cleaning the columns 'title' and 'snippet' as both will be needed to evaluate and predict the rank and both are strings so it needs to be clean. We then make a function 'clean_text'. In this function we first convert all the letters to lower case and then we will replace the all kinds of special characters with space , following with removing the numbers and symbols like forward and backward slash. Thereafter creating the function, it will be applied on the column 'title' and 'snippet' by the use of predefined function "apply" which help us to run the same code for all the rows in the column this will save the space and time complexity otherwise we have to use for loop to perform it on every rows.

After performing this,A new data frame created named 'fu' in which both the column is concatenated with the space and stored in the 'fu' data frame on the column name 'combined' shown in figure 12 . After concatenation we then tokenize the words to process

in the model. and in resultant we get an array. of datatype object.
Thereafter processing all the steps we are now ready to make the store the data in variable 'X' with predictable data and 'Y' with predicting data and then split it into train and test on the ratio of 90 and 10 respectively in figure 17.

3.1.5 Modeling

Now the data is ready to feed into the model. The Recurrent Neural Network architecture has been used where the LSTM model is applied (RNN and LSTM is described in design specification). nor the modelling process first the model proceed with sequential after that the embedding layer as been introduced, the model works with layers so we will have to add the layers, then the Spatial Drouout1d layer inserted below the embedding layer. After that the LSTM layer is added with 100 nodes and on dropout with .2 in the next step the dense layer is added it can be of any size but in the applied model it start with 10 nodes. the dense layer can be increased with the increasing of the nodes but it must be in the power of two like 2,4,8,16,32 etc. after processing this the model will be compile with the accuracy metrics and loss. both of them will be analysed by changing the weights.
After applying the model the accuracy will be calculated by the formula
accuracy = True positive/(True Positives + False Positives)

4 Design Specification

In the current era of technology, the neural network is gaining the growth in machine learning and artificial intelligence. We have performed the Recurrent Neural Network in our project. Our dataset is based on the classification and also in the form of string. So, the Recurrent Neural Network is the best option to perform this task and to complete this we need a classifier which we will be using Long Short-Term Memory (LSTM) in figure Figure 7.

Before applying neural network there are some steps which we will be processing, In the start we first introduced the data which is already in the .csv format and then proceed with the Exploratory Data Analysis where we get the insights of the data and try to understand how we can perform with it. After processing with EDA we take out the required column into another data-frame and also separating the target variable. Now we have our final data-frame for predicting the variable but before that we have to clean it as it has many impurities like Stop words 'like is, am, are, on, of etc it will remove them all and provides us the characteristic words', special characters 'like dollar, euro, pound, hash, percentile, forward or backward slash', numeric values. After cleaning this we tokenize the data and it formed an array which is now ready for splitting in test and train data. But before splitting the data it passes through sequential function and then the data split in 90:10 ratio where 90% will go for training and 10 will go for testing. Now we feed the data into model.
In the process of modelling first the data passes through the function called sequential where every layer has one output and input tensor. It if good for layers which is plain stack. Now we add another layer name Embedding. The embedding helps in converting the high dimensional to low dimensional vector. It makes easy to understand for modelling further because it shows vector continuously of discrete variables. Or in simpler form it converts the text into vectors which makes the data more understandable for the

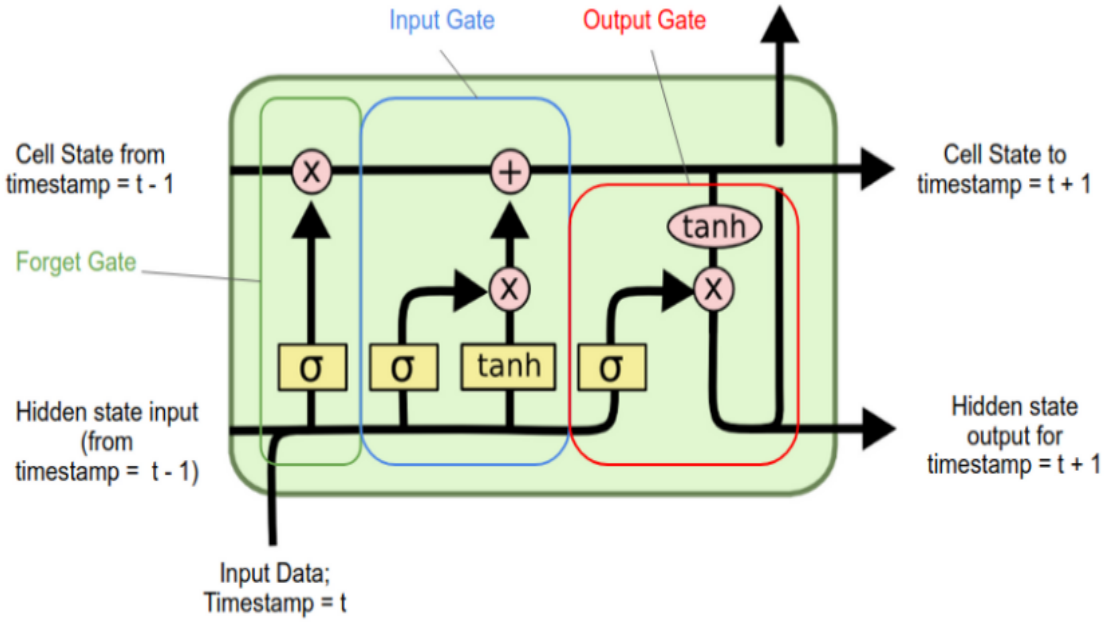


Figure 7: LSTM

model. After performing this now another layer inserted named Spatial Dropout1D, as the name says it works as 1d transition and dropout will drop it, now what it will do is, it won't drop the single elements instead of, it will drop whole feature maps of 1D. After this the layer of Long Short-Term Memory is inserted. This layer tries to learn the dependent variable for the target variable, it is used for predicting the sequence problem that's why it is used in the modelling. It also works very good in problems which are very complex and especially in form of text etc. It is a part of Recurrent Neural Network architecture. The major difference between the RNN and LSTM is that there is no cell state in RNN where has LSTM has both cell and hidden states. And because of this cell state it named long term memory altogether there is a loop which shows recursiveness. That means the output of every cell is the input of next cell which allows more effectiveness and help in the extraction of features in the text or input. It also increases the process of learning as increasing the cell to an extent.
Now before compiling the model only one more layer is added which is dense layer Figure 8. The dense layer is very common layer used in different applications of artificial intelligence and machine learning. It is connected enormously connected to output layer or to the next connecting layer. In this layer every single node connected to every node of next layer.
Now in the end the compilation of model executed and retrieve the results.

5 Implementation

This section will described the implementation of the model Long Short Term Memory of Recurrent Neural Network architecture. This models help to predict the classification

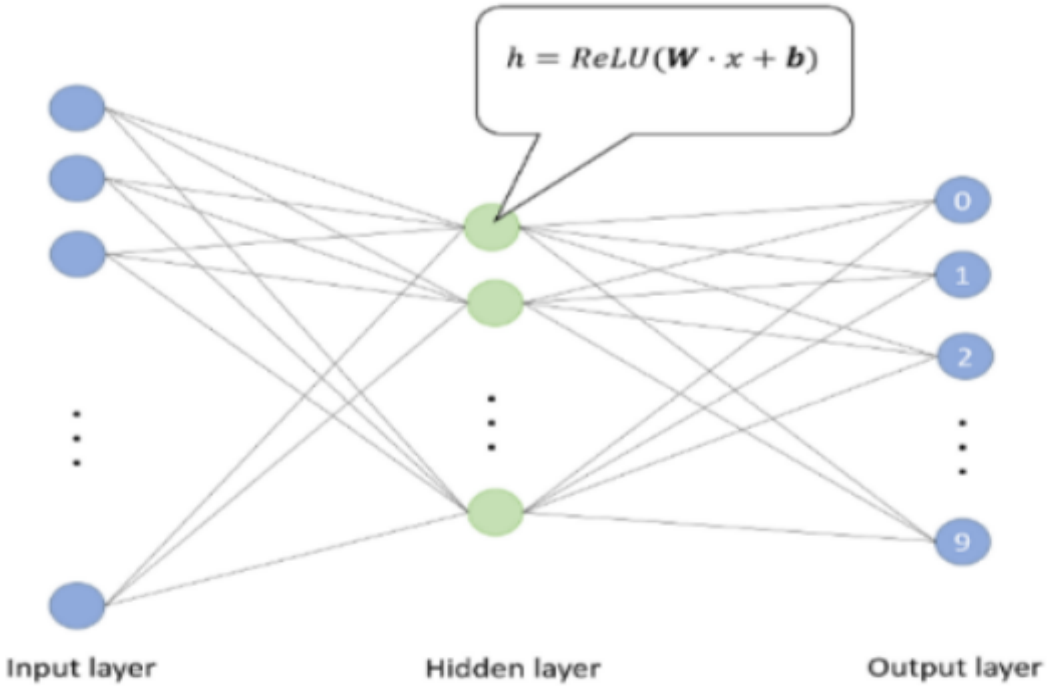


Figure 8: Dense Layer

of the rank on the search engine result page. Extraction of features has been identified with the help of various transfer learning models such as embedding system, spatial Droupout1D, Dense layers, LSTM layer in the RNN. The data in form of array which has be tokenized by the tokenizer. The Lstm is feed with 100 nodes and droupout with 20% with the same recurrent Dropout. it is followed with the Dense layer with the units of 10 and the function used is Softmax Activation. The model is been then compiled with loss in categorical cross entropy, optimizer with adam and metrics with accuracy. after performing these step we get the model summary which is shown in Figure 9 The epoch is 50 and the batch size is 25 so after fitting the model and getting the history we retrieve the result of accuracy is 0.4519 with the loss of 1.53. 10 epochs run with 130 times as the batch size is 25. The value loss is 2.2871 and Value accuracy is 0.2194. It started with the accuracy of 01485 and loss of 2.25 bus as increasing the epochs the accuracy is increased initially for 3 run it gives the good growth of 0.7 but after that growth start decreasing and giving the growth with 0.3 and as the epochs increasing the growth of accuracy decreases and in the end is it approx 0.13 so it stop running as because of the function Early Stopping have been applied with the value of patience 6.

5.1 Environmental Setup

To run this model and the RNN architecture we have used the system of 11th generation with intel core i5 which has 24GB of RAM and 512 SSD hard disk. The Anaconda software is used for launching the jupyter Notebook. the version of Python is 3.9. Keras is used for modeling which imported by installing the Tenserflow and the version is 2.7 which is written in Python C++ and CUDA. The whole task is performed on single language Python. The implementation of epoch in every model is 50 with the batch size

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, 250, 100)          5000000

 spatial_dropout1d (SpatialD  (None, 250, 100)         0
 ropout1D)

 lstm (LSTM)                 (None, 100)               80400

 dense (Dense)               (None, 10)                1010

=================================================================
Total params: 5,081,410
Trainable params: 5,081,410
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 9: Compiled Model

of 25. The early stopping Patience is 6 with minimum delta is 0.0001 and the model will me monitoring the value loss.

6 Evaluation

In this section of the paper the evolution of model has been described that how they have been improvised. The findings of the model.

6.1 Case Study 1

This is the first case where we have applied our first model with the column of 'title','snippet','meta data' and them then combined them with concatenation. This concatenation formed an array which will be used for tokenzing and provide us the array. Figure 10

```
Shape of data tensor: (4000, 250)
```

Figure 10: Shape of data tensor

After this we provide the predicting values to a variable and provide the shape. Figure 11

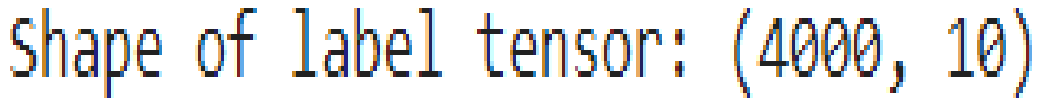


Figure 11: Shape of label tensor

Now we split the array in test and train with the 90% for tarining and 10% of testing shown in Figure 12

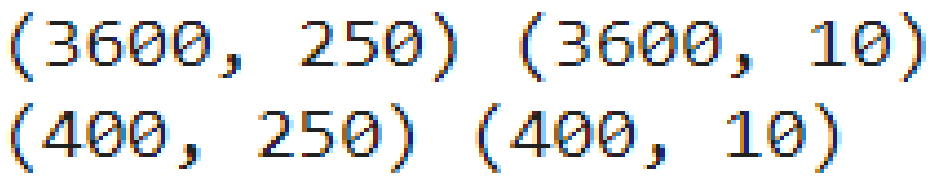


Figure 12: Test and Train shape

Now first model assigned as sequential and after that embedded layer is added following with Spatial Dropout1D layer prceeding with LSTM layer with 100 nodes and then the dense layer is added with 10 units and then compiled all the layers. The result of compilation shown in Figure 13

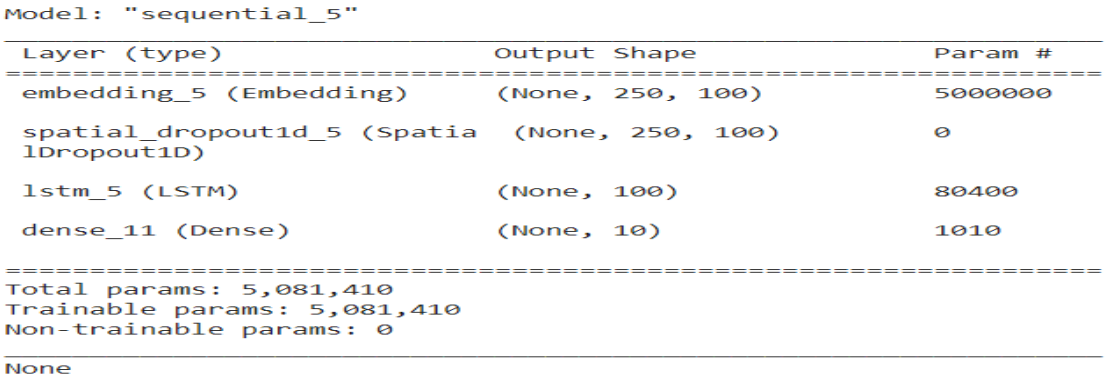


Figure 13: compiled model

Now everything is ready to fit the model. and after fitting the model with epoch size 50 and with the batch size of 25. Early stopping function has been used with the patience of 6 and minimum delta is assigned with 0.0001 and the result is shown in Figure 14

By applying the model with one dense layer , one lstm layer, one embeded layer, one Spatial layer we achieve the result of 43% after running 10epochs out of 50 because

```
Epoch 1/50
130/130 [==============================] - 82s 633ms/step - loss: 2.1335 - accuracy: 0.2102 - val_loss: 2.1028 - val_accuracy: 0.2250
Epoch 2/50
130/130 [==============================] - 81s 625ms/step - loss: 1.9486 - accuracy: 0.2787 - val_loss: 2.0871 - val_accuracy: 0.2333
Epoch 3/50
130/130 [==============================] - 81s 625ms/step - loss: 1.8523 - accuracy: 0.3056 - val_loss: 2.0781 - val_accuracy: 0.2694
Epoch 4/50
130/130 [==============================] - 82s 629ms/step - loss: 1.7894 - accuracy: 0.3355 - val_loss: 2.0460 - val_accuracy: 0.2583
Epoch 5/50
130/130 [==============================] - 81s 624ms/step - loss: 1.7183 - accuracy: 0.3580 - val_loss: 2.0971 - val_accuracy: 0.2389
Epoch 6/50
130/130 [==============================] - 82s 631ms/step - loss: 1.6735 - accuracy: 0.3688 - val_loss: 2.1366 - val_accuracy: 0.2583
Epoch 7/50
130/130 [==============================] - 79s 605ms/step - loss: 1.6365 - accuracy: 0.3948 - val_loss: 2.1516 - val_accuracy: 0.2444
Epoch 8/50
130/130 [==============================] - 81s 624ms/step - loss: 1.5967 - accuracy: 0.4019 - val_loss: 2.2404 - val_accuracy: 0.2722
Epoch 9/50
130/130 [==============================] - 81s 620ms/step - loss: 1.5505 - accuracy: 0.4198 - val_loss: 2.2897 - val_accuracy: 0.2694
Epoch 10/50
130/130 [==============================] - 81s 620ms/step - loss: 1.5122 - accuracy: 0.4343 - val_loss: 2.3023 - val_accuracy: 0.2861
```

Figure 14: Epoch history with result of accuracy

of early stopping function. so that it can be concluded that it is not good enough for preciseness.it started with the accuracy of 21% in first epoch but in the last by performing 10 epochs cycles it gives only 43%.

6.2 Case Study 2

Now to achieve more accuracy another model is applied but before that we have changed the strategy instead of taking three columns lets take only two columns that are 'title' and 'snippet'. it will follow the same procedure as above till tokenizer. In the formation of model the dense layer has been added and in the previous model to check the impact on the accuracy. In the first dense layer the units provided are 256 following with 64 and 32 with the Activation function of RELU and in the end 10 units assigent to the dense layer with the activation function of Softmax. Now the model has been compiled and the result is shown in Figure 15

```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 embedding_3 (Embedding)      (None, 250, 100)          5000000

 spatial_dropout1d_3 (Spatia  (None, 250, 100)          0
 lDropout1D)

 lstm_3 (LSTM)                (None, 100)               80400

 dense_6 (Dense)              (None, 256)               25856

 dense_7 (Dense)              (None, 64)                16448

 dense_8 (Dense)              (None, 32)                2080

 dense_9 (Dense)              (None, 10)                330

=================================================================
Total params: 5,125,114
Trainable params: 5,125,114
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 15: Compiled model

With this experiment, the accuracy achieved is 40% only shown in Figure 16 which is lesser that the previous one. so this model is not useful to predict the ranking of web page.

```
Epoch 1/50
130/130 [==============================] - 83s 605ms/step - loss: 2.2734 - accuracy: 0.1417 - val_loss: 2.1804 - val_accuracy: 0.1889
Epoch 2/50
130/130 [==============================] - 80s 615ms/step - loss: 2.1605 - accuracy: 0.1941 - val_loss: 2.1270 - val_accuracy: 0.2083
Epoch 3/50
130/130 [==============================] - 80s 619ms/step - loss: 2.0630 - accuracy: 0.2102 - val_loss: 2.1238 - val_accuracy: 0.2028
Epoch 4/50
130/130 [==============================] - 80s 612ms/step - loss: 1.9829 - accuracy: 0.2500 - val_loss: 2.1047 - val_accuracy: 0.2111
Epoch 5/50
130/130 [==============================] - 80s 616ms/step - loss: 1.9133 - accuracy: 0.2880 - val_loss: 2.1267 - val_accuracy: 0.2111
Epoch 6/50
130/130 [==============================] - 78s 597ms/step - loss: 1.8424 - accuracy: 0.3210 - val_loss: 2.1501 - val_accuracy: 0.2000
Epoch 7/50
130/130 [==============================] - 77s 595ms/step - loss: 1.7894 - accuracy: 0.3321 - val_loss: 2.1751 - val_accuracy: 0.2139
Epoch 8/50
130/130 [==============================] - 77s 592ms/step - loss: 1.7269 - accuracy: 0.3617 - val_loss: 2.2256 - val_accuracy: 0.2083
Epoch 9/50
130/130 [==============================] - 77s 590ms/step - loss: 1.6851 - accuracy: 0.3830 - val_loss: 2.2711 - val_accuracy: 0.2306
Epoch 10/50
130/130 [==============================] - 77s 595ms/step - loss: 1.6310 - accuracy: 0.4003 - val_loss: 2.2734 - val_accuracy: 0.2250
```

Figure 16: Epoch history with result of accuracy

6.3 Case Study 3

The above two case studies are failed,so the model again formed with same RNN architecture with the help of LSTM layer of same nodes as above used but here we have removed one dense layer that is of unit 128. Now the model proceed with three dense layer that are of unit 64, 32, 10. In which the Dense layer with unit 64 and 32 are having the activation funtion of RELU and the other one with 10 unit is performed with the activation fumction of Softmax. The compiled model is shown in Figure 17

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_2 (Embedding)     (None, 250, 100)          5000000

 spatial_dropout1d_2 (Spatia  (None, 250, 100)         0
 lDropout1D)

 lstm_2 (LSTM)               (None, 100)               80400

 dense_3 (Dense)             (None, 64)                6464

 dense_4 (Dense)             (None, 32)                2080

 dense_5 (Dense)             (None, 10)                330

=================================================================
Total params: 5,089,274
Trainable params: 5,089,274
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 17: compiled model

Now with the same procedure of fitting the model it has been fitted and gives the accuracy of 41.48% is shown in Figure 18 which is 1% greater than the above model but lesser than the case study 1. so there is something more can be done to enhance the accuracy of the model. in the next case study the another model has been applied. In this model the the first cycle of epochs run and give the accuracy of 14% only but it start increasing by every step and after eleven cycles of epochs it achieve the accuracy of 41% also another thing is that in the above experiment on;y 10 cycles of epochs run but in this eleven cycles run.

6.4 Case Study 4

In this experiment another model is organised with the same RNN architecture with LSTM layers but the dense layers has been removed which has the activation function

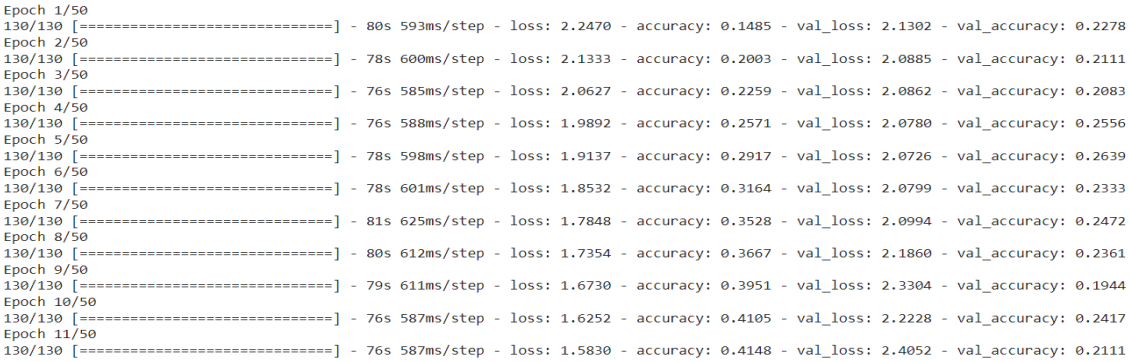


Figure 18: Epoch history with result of accuracy

RELU that are having 64 and 32 units where as we keep only one layer with the unit of 10 and has activation function Softmax. we then complied the model with only embedded layer, LStm layer, Spatial Dropout1D layer, and one dense layer. the compiled model is shown in Figure 19. The size of epoch is 50 but the batch size is reduced to 16

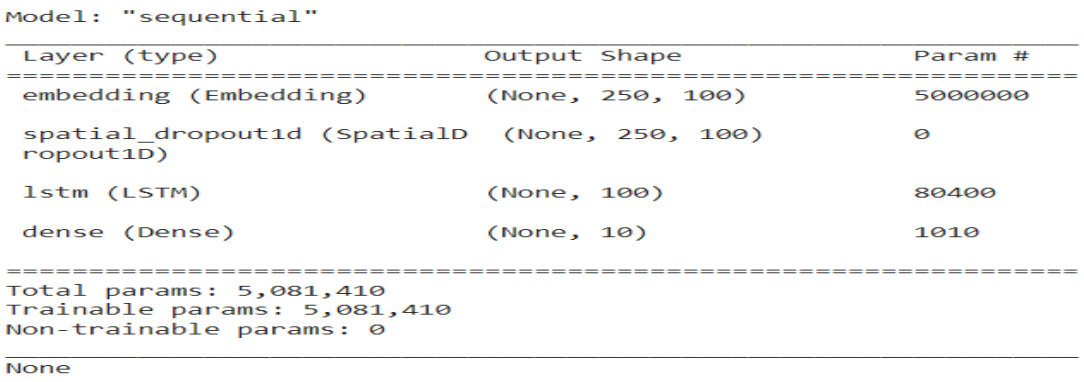


Figure 19: Model compiled

After compiling the model we have fitted the model and retrieve the accuracy of 55% which is the highest among the all of the experiments.In this cycle the epochs have run 16 cycle with start of48% in the beginning and following it it goes till 55% with the loss of 1.05. as shown in Figure 20

6.5 Case Study 5

This is our last case study where the model is as used in the first case study where the model aceive the accuracy of 43% so and other prceeding model we gan the accuracy till 55% so try another method by changing the weights of epochs and batch sizes and early stopping patience by 50 , 16, 15 respectively. As from the result after removing one column the model is getting the good accuracy. By changing this we gained the accuracy of approx 64% with the loss of 0.9. is is very efficient as compared to other model. The result is sown in Figure 21

```
Epoch 1/50
203/203 [==============================] - 104s 511ms/step - loss: 1.3470 - accuracy: 0.4827 - val_loss: 2.6894 - val_accuracy: 0.2194
Epoch 2/50
203/203 [==============================] - 104s 510ms/step - loss: 1.3112 - accuracy: 0.4923 - val_loss: 2.7699 - val_accuracy: 0.1972
Epoch 3/50
203/203 [==============================] - 104s 511ms/step - loss: 1.2781 - accuracy: 0.5040 - val_loss: 2.7483 - val_accuracy: 0.2139
Epoch 4/50
203/203 [==============================] - 103s 507ms/step - loss: 1.2567 - accuracy: 0.4954 - val_loss: 2.8426 - val_accuracy: 0.2167
Epoch 5/50
203/203 [==============================] - 103s 507ms/step - loss: 1.2328 - accuracy: 0.5120 - val_loss: 2.8876 - val_accuracy: 0.2083
Epoch 6/50
203/203 [==============================] - 103s 508ms/step - loss: 1.2101 - accuracy: 0.5123 - val_loss: 2.8326 - val_accuracy: 0.2083
Epoch 7/50
203/203 [==============================] - 103s 507ms/step - loss: 1.1864 - accuracy: 0.5207 - val_loss: 3.0918 - val_accuracy: 0.1861
Epoch 8/50
203/203 [==============================] - 104s 512ms/step - loss: 1.1715 - accuracy: 0.5299 - val_loss: 3.0606 - val_accuracy: 0.2194
Epoch 9/50
203/203 [==============================] - 104s 513ms/step - loss: 1.1619 - accuracy: 0.5235 - val_loss: 3.1532 - val_accuracy: 0.2028
Epoch 10/50
203/203 [==============================] - 104s 513ms/step - loss: 1.1310 - accuracy: 0.5392 - val_loss: 3.1821 - val_accuracy: 0.2167
Epoch 11/50
203/203 [==============================] - 104s 514ms/step - loss: 1.1193 - accuracy: 0.5497 - val_loss: 3.0888 - val_accuracy: 0.2417
Epoch 12/50
203/203 [==============================] - 104s 515ms/step - loss: 1.1087 - accuracy: 0.5460 - val_loss: 3.2288 - val_accuracy: 0.1944
Epoch 13/50
203/203 [==============================] - 104s 514ms/step - loss: 1.0976 - accuracy: 0.5441 - val_loss: 3.2457 - val_accuracy: 0.2250
Epoch 14/50
203/203 [==============================] - 104s 514ms/step - loss: 1.0954 - accuracy: 0.5401 - val_loss: 3.2244 - val_accuracy: 0.2194
Epoch 15/50
203/203 [==============================] - 105s 515ms/step - loss: 1.0699 - accuracy: 0.5485 - val_loss: 3.3719 - val_accuracy: 0.2222
Epoch 16/50
203/203 [==============================] - 104s 515ms/step - loss: 1.0563 - accuracy: 0.5531 - val_loss: 3.4700 - val_accuracy: 0.2167
```

Figure 20: Epoch history with result of accuracy

```
Epoch 1/50
203/203 [==============================] - 107s 526ms/step - loss: 1.2343 - accuracy: 0.5407 - val_loss: 2.5118 - val_accuracy: 0.2639
Epoch 2/50
203/203 [==============================] - 107s 526ms/step - loss: 1.2155 - accuracy: 0.5534 - val_loss: 2.5942 - val_accuracy: 0.2611
Epoch 3/50
203/203 [==============================] - 107s 525ms/step - loss: 1.2004 - accuracy: 0.5531 - val_loss: 2.6062 - val_accuracy: 0.2722
Epoch 4/50
203/203 [==============================] - 106s 523ms/step - loss: 1.1776 - accuracy: 0.5611 - val_loss: 2.5796 - val_accuracy: 0.2639
Epoch 5/50
203/203 [==============================] - 107s 525ms/step - loss: 1.1262 - accuracy: 0.5843 - val_loss: 2.6682 - val_accuracy: 0.2806
Epoch 6/50
203/203 [==============================] - 106s 523ms/step - loss: 1.1224 - accuracy: 0.5812 - val_loss: 2.6335 - val_accuracy: 0.2694
Epoch 7/50
203/203 [==============================] - 106s 524ms/step - loss: 1.1131 - accuracy: 0.5855 - val_loss: 2.6950 - val_accuracy: 0.2667
Epoch 8/50
203/203 [==============================] - 107s 528ms/step - loss: 1.0705 - accuracy: 0.6006 - val_loss: 2.8312 - val_accuracy: 0.2639
Epoch 9/50
203/203 [==============================] - 108s 530ms/step - loss: 1.0614 - accuracy: 0.5920 - val_loss: 2.8828 - val_accuracy: 0.2694
Epoch 10/50
203/203 [==============================] - 107s 527ms/step - loss: 1.0426 - accuracy: 0.6130 - val_loss: 2.8031 - val_accuracy: 0.2778
Epoch 11/50
203/203 [==============================] - 107s 526ms/step - loss: 1.0445 - accuracy: 0.6000 - val_loss: 2.9189 - val_accuracy: 0.2750
Epoch 12/50
203/203 [==============================] - 106s 524ms/step - loss: 1.0034 - accuracy: 0.6167 - val_loss: 2.8528 - val_accuracy: 0.2833
Epoch 13/50
203/203 [==============================] - 106s 524ms/step - loss: 0.9998 - accuracy: 0.6228 - val_loss: 2.9885 - val_accuracy: 0.2750
Epoch 14/50
203/203 [==============================] - 107s 525ms/step - loss: 0.9869 - accuracy: 0.6194 - val_loss: 2.9676 - val_accuracy: 0.2750
Epoch 15/50
203/203 [==============================] - 107s 527ms/step - loss: 0.9713 - accuracy: 0.6238 - val_loss: 2.9966 - val_accuracy: 0.2750
Epoch 16/50
203/203 [==============================] - 107s 529ms/step - loss: 0.9330 - accuracy: 0.6392 - val_loss: 3.1454 - val_accuracy: 0.2611
```

Figure 21: Epoch history with result of accuracy

6.6 Discussion

The model which is applied in the project is achieved the highest accuracy of 64% in case study 5 where as our other models didn't perform well. The results from the other models we get the accuracy of 40%, 41%, 45% and 43% but their are more work to do. such as the data in the data set is not sufficient. The data set is more useful for analyzing the insights and on the other hand the data is dynamic which means it is scraped fro real world search engine not from a particular organisation which provides all the details related to the search ranking but the data set doesn't include that much information for predicting rank such as no. of visitors on the web page etc. This data set only includes string values where the model extract the keywords and features. Through this extraction they have performed much better. In the previous research they have performed the value on the keywords that how much they give the impact on the ranking regardless predicting them. Another research is on the data set of an e-commerce organisation where they get good results but they also said that their model is not enough capable of real world search engine. but our model gives the result on real data set scraped from the google search engine and provide the efficiency of 64%.

7 Conclusion and Future Work

Our research question is to predict the ranking of the web page based on the data stored behind the web page like title of the web page, snippets, meta data etc. by applying the machine learning models and deep learning techniques.
We have succeed to in prediction and achieved the accuracy of 64% by applying the LSTM from RNN architecture with the batch size of 16 and epochs 50. Fur layer have been compiled to create a model to perform the task. It could be more accurate and precise if the data set include the no. of visitors on the web page, no. of clicks on the web page, total no. of time every visitor spend on the web page. Through some of these factors we may be able to predict the ranking more accurate.
In the future work the textual data can be embedded more precisely, cosine vectorize can be used and in the modeling for more feature extraction. And with the more data and resources the Reinforcement learning model can be applied for predicting the ranking of web page.

References

Arora, P. and Bhalla, T. (2014). A synonym based approach of data mining in search engine optimization, *International Journal of Computer Trends and Technology* **12**(4): 201–205.

Chauhan, V., Jaiswal, A. and Khan, J. (2015). Web page ranking using machine learning approach, *2015 Fifth International Conference on Advanced Computing Communication Technologies*, pp. 575–580.

Cunningham, S. (1998). Dataset cataloging metadata for machine learning applications and research.

Drivas, I., Sarlis, A. and Sakas, D. (2017). Implementation and dynamic simulation modeling of search engine optimization processes. improvement of website ranking.

Eirinaki, M., Vazirgiannis, M. and Kapogiannis, D. (2005). Web path recommendations based on page ranking and markov models, WIDM '05, Association for Computing Machinery, New York, NY, USA, p. 2–9.
URL: *https://doi.org/10.1145/1097047.1097050*

Evans, M. P. (2007). Analysing google rankings through search engine optimization data, *Internet Research* **17**(1): 21–37.

Hu, Y., Da, Q., Zeng, A., Yu, Y. and Xu, Y. (2018). Reinforcement learning to rank in e-commerce search engine, *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining* .

Khekare, G. and Verma, P. (2020). Design of automatic key finder for search engine optimization in internet of everything, *2020 IEEE 1st International Conference for Convergence in Engineering (ICCE)*, pp. 464–468.

Khorsheed, K., Madbouly, M., Guirguis, S., Khorsheed, K., Madbouly, M. and Guirguis, S. (2015). Search engine optimization using data mining approach, **IX**: 184.

Kumar, S., Gao, X., Welch, I. and Mansoori, M. (2016). A machine learning based web spam filtering approach, *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, pp. 973–980.

Lee, S., Jang, W., Lee, E. and Oh, S. G. (2016). Search engine optimization, *Library Hi Tech* **34**(2): 197–206.

Malaga, R. A. (2010). Chapter 1 - search engine optimization—black and white hat approaches, *Advances in Computers: Improving the Web*, Vol. 78 of *Advances in Computers*, Elsevier, pp. 1–39.
URL: *https://www.sciencedirect.com/science/article/pii/S0065245810780013*

Mandl, T. (2006). Implementation and evaluation of a quality-based search engine, *Proceedings of the Seventeenth Conference on Hypertext and Hypermedia*, HYPERTEXT '06, Association for Computing Machinery, New York, NY, USA, p. 73–84.
URL: *https://doi.org/10.1145/1149941.1149957*

Sagot, S., Fougères, A.-J., Ostrosi, E. and Lacom, P. (2014). Search engine optimization: From analysis based on an engineering meta-model towards integrative approaches, *International Conference on Information Society (i-Society 2014)* pp. 274–281.

Salminen, J., Corporan, J., Marttila, R., Salenius, T. and Jansen, B. J. (2019). Using machine learning to predict ranking of webpages in the gift industry, *Proceedings of the 9th International Conference on Information Systems and Technologies* .

Sen, R. (2005). Optimal search engine marketing strategy, *International Journal of Electronic Commerce* **10**(1): 9–25.
URL: *https://doi.org/10.1080/10864415.2005.11043964*

Singh, R. and Gupta, S. (2013). Search engine optimization - using data mining approach.
URL: *https://www.ijaiem.org/volume2issue9/IJAIEM-2013-09-16-015.pdf*

Wang, F., Li, Y. and Zhang, Y. (2011). An empirical study on the search engine optimization technique and its outcomes, *2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC)*, pp. 2767–2770.

Zhang, J. and Dimitroff, A. (2005). The impact of metadata implementation on webpage visibility in search engine results (part ii), *Information Processing Management* **41**(3): 691–715. Cross-Language Information Retrieval.
URL: *https://www.sciencedirect.com/science/article/pii/S0306457303001134*