TEXT SUMMARIZATION USING NLP

TEAM - 2

201 S High Point Road APT 213 Madison, Wisconsin – 53717 USA
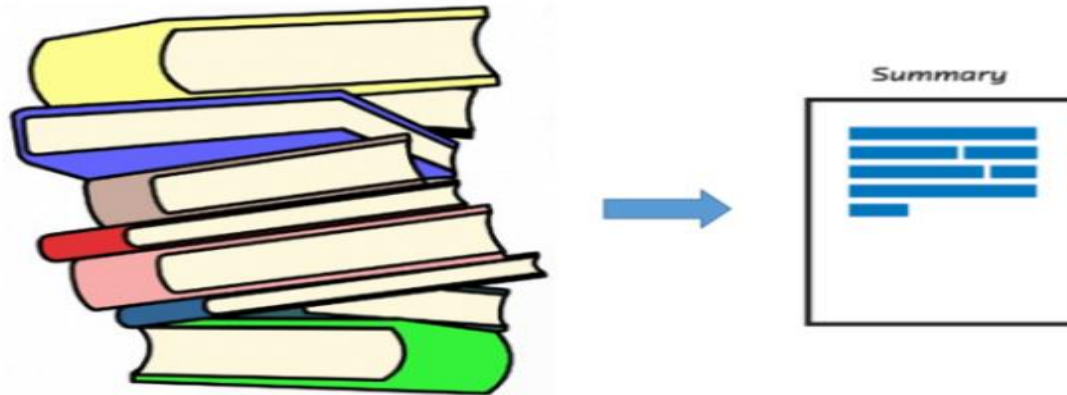
ADITHYA HARSHA   ,   CHETHAN KUMAR

MENTOR :- MUNMUN Ma'am, EXCEL R

DAT E :- 24-05-2022

INTRODUCTION :-

- The Goal of Summarization is to produce a shorter version of a source text by preserving the meaning and the key contents of the original Book.

- A well written summary can significantly reduce the amount of work needed to digest large amount of text.

BUSINESS PROBLEM :-

Text summarization is one of the most interesting problems in NLP. It's hard for us, as humans, to manually extract the summary of a large document of text. To solve this problem, we use automatic text summarization. It's a way of identifying meaningful information in a document and summarizing it while conserving the overall meaning.

OBJECTIVE :-

The purpose is to present a shorter version of the original text while preserving the semantics. Here, you can use different traditional and advanced methods to implement automatic text summarization, and then compare the results of each method to conclude which is the best to use for your corpus.
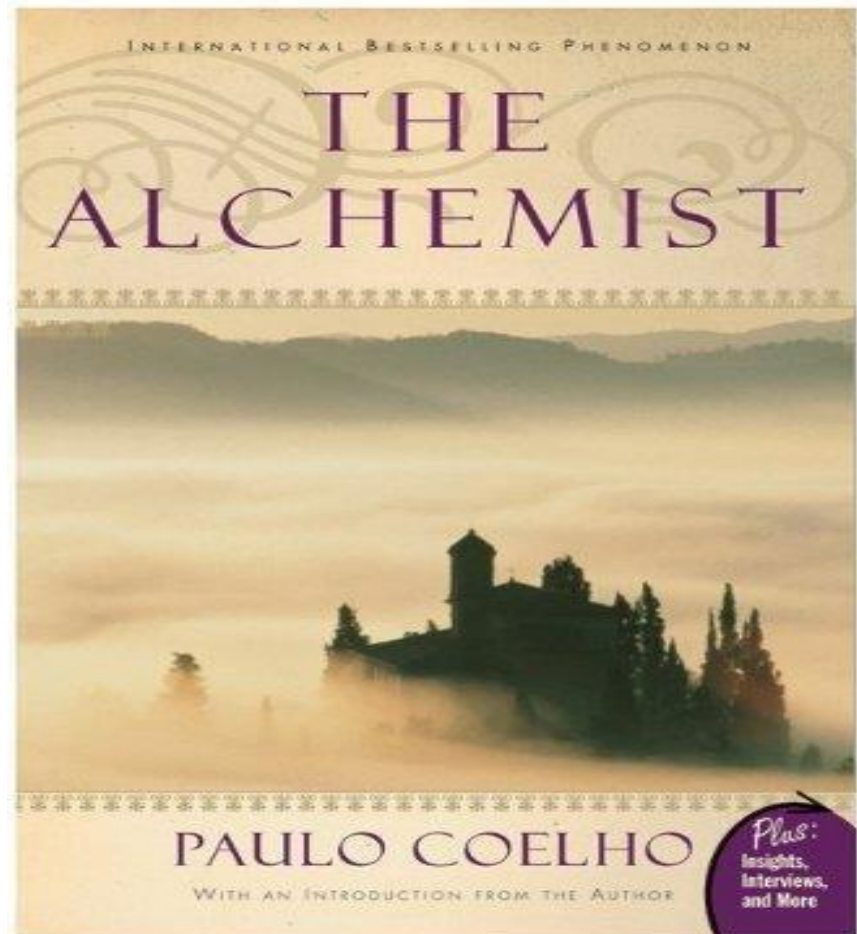
DATA INFORMATION : -

Book Name :- The Alchemist

Author :- Paul Coelho

Source :- Google

Pages :- 158

PROJECT ARCHITECTURE :-

EXPLORATORY DATA ANALYSIS (EDA)

- EDA is the basic step of the project. Before performing operations on the data, we need to clean the data.

- In text summarization , EDA can be done to remove the stopwords and punctuations in the text data.

- EDA Consists of following steps :-

    * Tokenization

    * Stopwords Removal

    * Punctuation Removal

    * Stemming

    * Lemmatization

TOKENIZATION

- Tokenization is used to divide the text data into words or sentences.

- Tokenization is done by two ways :-

  a) Word based tokenization

  b) Sentence based tokenization

- We tokenized our data in the form of word based tokenization.

- We are using the NLTK library of python.

PUNCTUATIONS REMOVAL

- For data processing data should be cleaned so we need to remove the punctuation from the data.

- Punctuations are often unnecessary as it doesn't add value or meaning.

STOPWORDS REMOVAL

- Stop words are most common words in any language that do not carry any meaning and are usually ignored by NLP

- Examples of stop words :-

    ON ,A ,THE ,IS ,ARE ,I ,FOR ,YOU ,AND ,IN

FEATURE EXTRACTION TECHNIQUE

- Feature extraction techniques are used to convert text into a vecor.

- It assigns different weights to words.

- Types of feature extraction techniques are as follows :-

  a) Bag of words (BOW)

  b) TF-IDF

  c) Word Embedding

STEMMING & LEMMATIZATION

- Stemming and lemmatization both are text reduction techniques.

- Using stemming and lemmatization we reduce the text can size.

- In stemming the suffix and prefix of the word is removed irrespective of meaning.

- In Lemmatization the suffix and prefix of the word is removed by maintaining the semantic of the word.

NAMED ENTITY RECOGNITION(NER)

alchemist paulo coelho translate by **alan r clarke** `PERSON` content introduction remember receive letter **american** `NORP` publisher harper collin ... prologue the alchemist pick book someone ... one the boy ' name santiago dusk fall ... **two** `CARDINAL` the boy work crystal merchant ... epilogue church night ... about the author international acclaim book by paulo coelho credit cover copyright about the publisher **ten year** `DATE` on remember receive a letter from the **american** `NORP` publisher harper collin say " read the alchemist like get dawn see sun rise rest world still sleep " struggle establish writer follow path despite voice tell impossible **brazilian** `NORP` journalist phone say president **clinton roberts** `PERSON` declare adore book walk alone the alchemist " what ' secret behind huge success god choose earth whenever something ' courage confront dream why there **four** `CARDINAL` obstacle first tell childhood onward everything want impossible we grow call deeply bury soul invisible but ' still afraid hurt around we abandon everything order pursue dream we realize love we realize genuinely wish we well want we happy prepare accompany we journey **third** `ORDINAL` obstacle fear defeat meet ' really want anyway " we want know stake everything path personal call patience difficult time know universe ask defeat necessary eight time go suffer people —

language would something like ' it write ' " ' way hold back river top but see crystal shop offer refreshing beautiful crystal glass would impress beauty glassware the man remark tea always delicious serve tradition orient use crystal glass tea magical power climb hill see shop something new trade old other shop open serve tea business shop seek man woman thirst thing new the boy awake before dawn it have be **eleven month nine day** `DATE` since **first** `ORDINAL` set foot **african** `NORP` continent **especially day** `DATE` he put headcloth place secure descend stair silently the city still sleep he prepare sandwich drink hot tea crystal glass then sit sunfilled sound wind bring scent desert when it bundle money enough buy **hundred** `CARDINAL` **africa** `LOC` country he wait patiently merchant **awaken** `PERSON` open shop then **two** `CARDINAL` go tea sheep and money need go mecca " the old man say nothing " will give blessing " ask boy " you then turn boy crystal shop but know ' go go mecca just know ' go buy sheep " and give boy blessing the boy go to his room and pack his belonging they fill **three** `CARDINAL` thummim realize long since think for nearly aside enough money could return **spain** `GPE` pride omen " strange sensation old king nearby he work hard thought even though sheep ' teach speak arabic but sheep teach something even important

BAG OF WORDS (BOW)

- BOW is the simple representative of the words.

- Representation of text describes the occurrence of words within the document.

- Generally it count the frequency of words that are repeated in the given document.

TERM FREQUENCY- INVERSE DOCUMENT FREQUENCY(TF-IDF)

- TF calculates the frequency of the word in given document divided by the total words in the document.

- **TF = No. of words in the document / Total count of words in the document.**

- IDF calculates the frequency of the words on the total document.

- **IDF = Total no. of documents / No. of documents with word in it.**

- Weight= TF*IDF

- So in this way weights are assigned to the words & is more semantic than the BOW.

- Higher weight is assigned to the most repetitive words so it is considered as less significant.

WORD EMBEDDING

- Word embedding is used for the representation of the word for text analysis.

- The most used type of word embedding is Word2Vec.

- Each word is basically represent as vector of 32 or more dimension instead of the single word.

- Here semantic information & relation between word is also preserved.

- Word2Vec assigns nearby vector according to the class.

WORD CLOUD
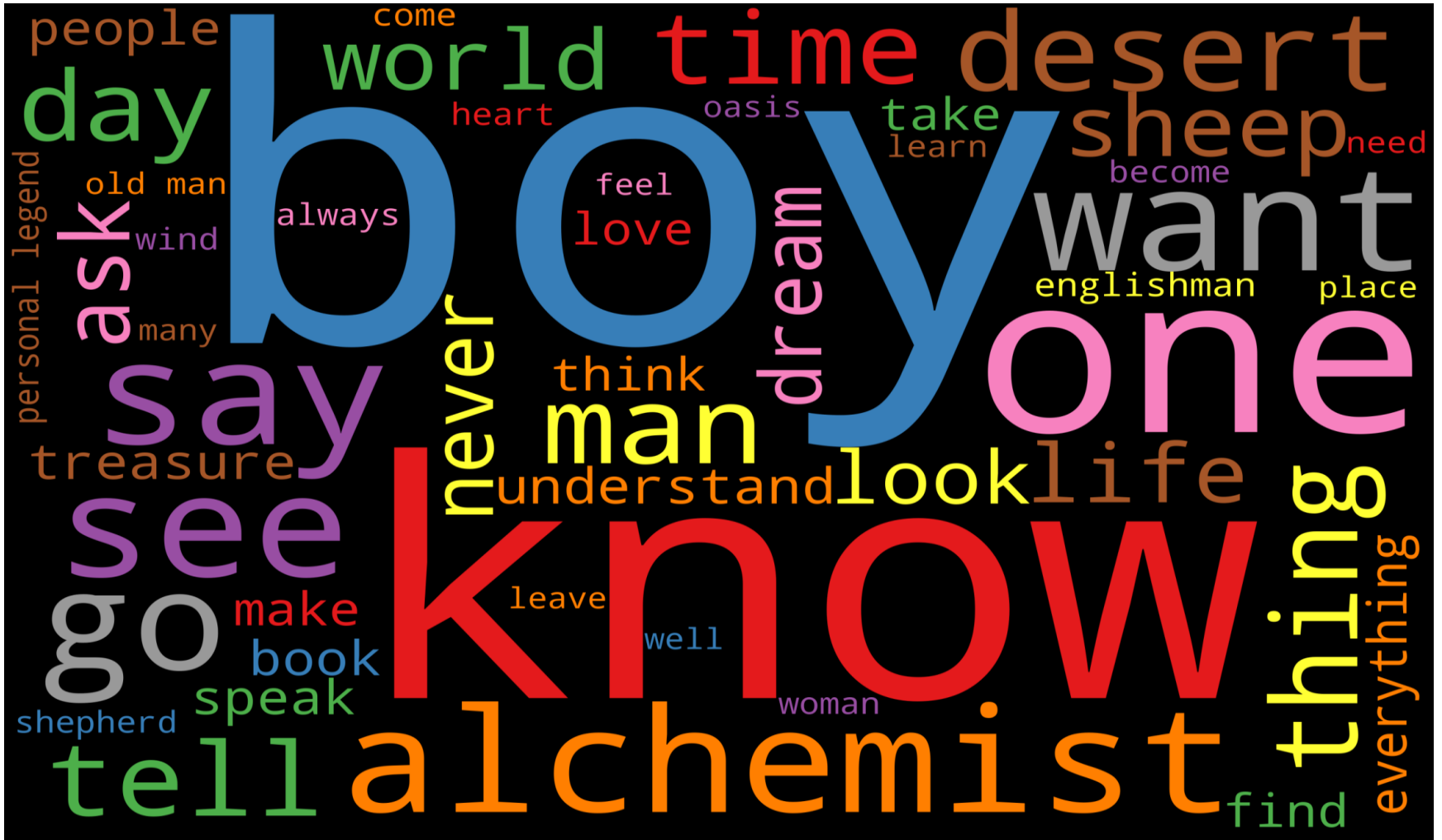
VISUALISING BAR CHAT TOP 10 Nouns + Verbs

Top 10 nouns and verbs
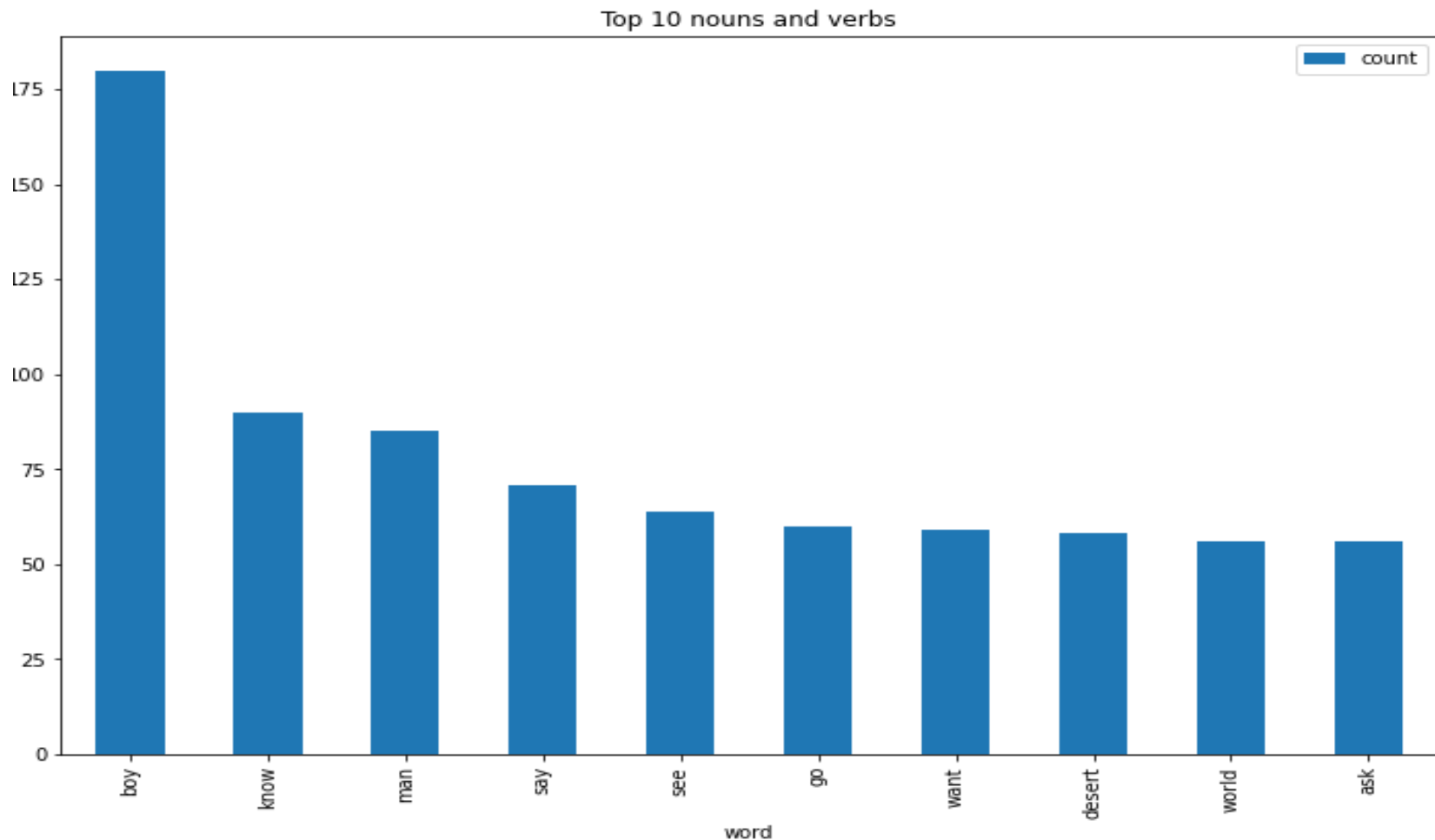
MODEL BUILDING

- We used pretrained models for text summarization and they are as follows :-

        1) Lex rank

        2) Luhn model

        3) LSA model

        4) Text rank

        5) Bart model

TEXT SUMMARIZATION

Text summarization are of two types

1) Extractive Text summarization :-

Extractive summarization picks up sentences directly from the document based on a scoring function to form a coherent summary. This method word by identifying important sections of the text and crops out and stitches together portions of the content to produce a condensed version.

Example :- Text Rank, Luhn, LexRank , LSA etc.

2) Abstractive Text summarization :-

Abstractive summarization methods aims at producing summary by interpreting the text using advanced natural language techniques in order to generate a new shorter text parts of which may not appear as part of the original document. Abstractive text summarization generates text with respect to the surrounding text.

Example :- LSTM, RNN, BERT, Transformer BART etc.

LEX RANK

LexRank is an unsupervised graph based approach for automatic text summarization.

In this model we have a connectivity matrix based on intra-sentence cosine similarity which is used as the adjacency matrix of the graph representation of sentences.

With the help of cosine similarity scored sentences are extracted from the document and are arranged in order.

<u>LUHN TEXT SUMMARIZATION</u>

Luhn's algorithm is a naive approach based on TF-IDF and on "window size" of non-important words and between words of high importance.

It also assigns higher weights to sentences occurring near the beginning of a document.

It is useful when very low frequent words as well as highly frequent words(stop words) both are not significant. Based on this, sentence scoring is carried out and high ranking sentences

make it to the summary.

LSA TEXT SUMMARIZATION

Latent Semantic Analysis is an unsupervised learning algorithm that

can be used for extractive text summarization. It extracts

semantically significant sentences by applying singular value

decomposition (SVD)

to the matrix of term-document frequency.

TEXT RANK

TextRank uses an extractive approach and is an unsupervised graph based text summarization technique. PageRank is an algorithm used to calculate rank of web pages, and is used by search engines such as Google. TextRank is based on the PageRank Algorithm.

It finds the similarities between sentences and then organizes in descending order.

BART MODEL

BART is a sequence - to - sequence model trained as a denoising autoencoder. This means that a fine - tuned BART model can take a text sequence as input and produce a different text sequence at the output. The idea here will be to use all the weights of the pretrained neural network model and use it as a initial point, in order to speed up training and improve performance.

MODEL EVALUATION

ROUGE-N MEASURE

Rouge measure the number of common N-grams between the generated summary and the original text.

With the help of this we measure PRECISION, RECALL and FI-Score

MODEL EVALUATION RESULT

Score	LexRank	Luhn	LSA	TextRank	BART
Recall	0.08484	0.11191	0.15117	0.06859	0.0909
Presicion	1	1	1	1	0.2647
F-score	0.1564	0.2012	0.2626	0.1283	0.1353

Conclusion :- LSA model gives high FI-Score we choose LSA model for deployment.

- LSA
- 0.15117
- 1
- 0.2626

MODEL DEPLOYMENT

- With the help of ROUGE score we choose our LSA model for the deployment. As LSA model gives high FI-Score value i.e 0.2626.

- We deployed our model with the help of streamlit.

- Streamlit is an open-source app framework for creating and deploying data science applications.

- We deploy our model in anaconda prompt.

MODEL DEPLOYMENT

MODEL DEPLOYMENT

MODEL DEPLOYMENT

CHALLENGES FACED ?

1. Removing Unnecessary things from the text file

2. Choosing the better model

HOW DID WE OVERCOME

1. Exploratory data analysis

2. Compare the accuracy of all the models

THANK YOU