CAPSTONE PROJECT

TED TALK VIEWS PREDICTION

TEAM MEMBERS

ANKIT KUMAR
FAHAD MEHFOOZ
SANJOG MISHRA
VARUN NAYYAR

CONTENTS

- INTRODUCTION
- PROBLEM STATEMENT
- METHODOLOGY
    1. DATA ACQUISITION
    2. EXPLORATORY DATA ANALYSIS AND DATA CLEANING
    3. TRAIN - TEST SPLIT
    4. DATA PREPROCESSING
    5. FEATURE ENGINEERING
    6. DATA MODELLING
    7. MODEL INTERPRETATION
    8. TRAIN AND TEST INTERPRETATION
- CONCLUSION
- CHALLENGES FACED
- FUTURE SCOPE OF WORK

INTRODUCTION

TED(Technology, Entertainment, Design) is a nonprofit devoted to spreading ideas, usually in the form of short, powerful talks. It is an American media organization that posts talks online for free distribution under the slogan “ ideas worth spreading”. These talks address a wide range of topics within the research and practice of science and culture, often through storytelling.

The notable programs and initiatives of TED include TED Talks, TED Conferences, TED Translators, TED-Ed.

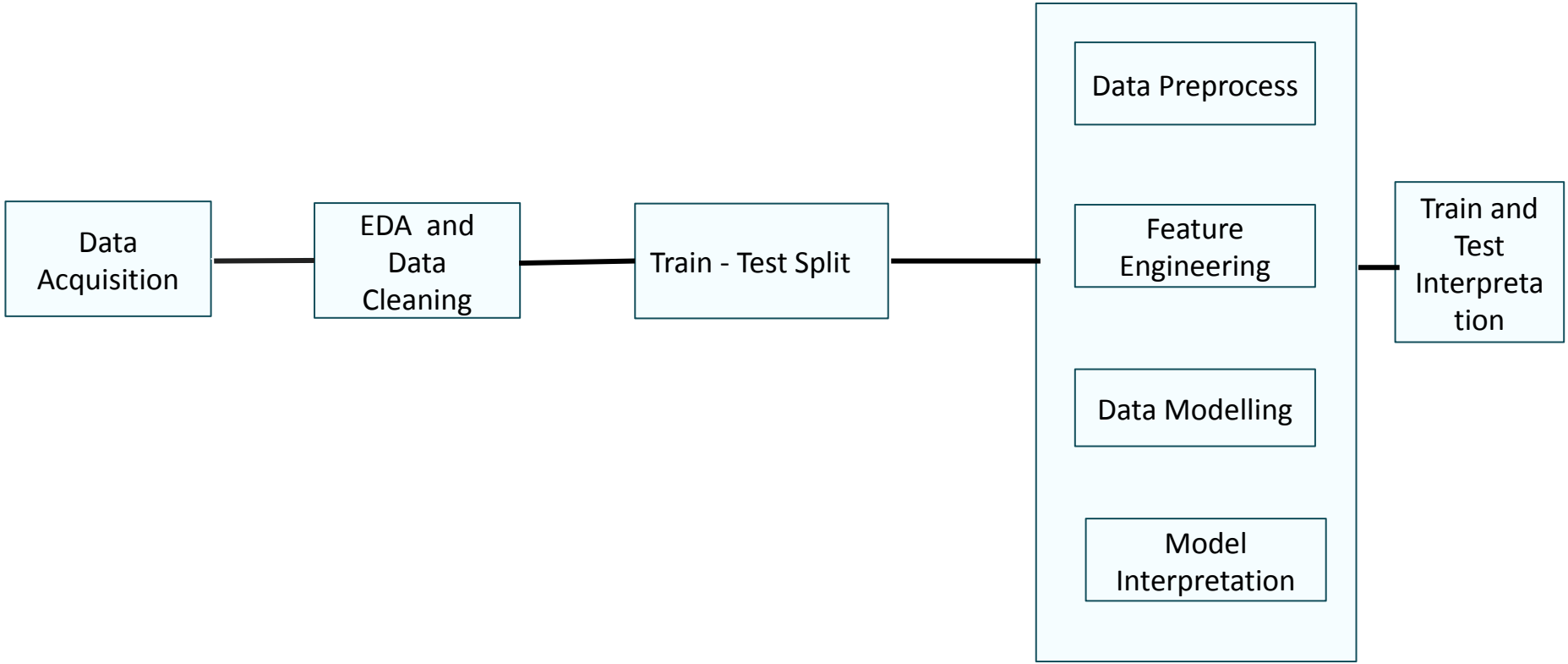
TED Ideas worth spreading

PROBLEM STATEMENT

Our objective is to predict the views of a TED talk uploaded on the TEDx website. For this purpose; we have been provided with the following attributes which are to be used to predict the views on a particular video.

- Information about the Speakers
- Underlying topics of the video's talk
- Type of Event in which the video was recorded
- Recorded and Published Date of the video
- Native language of the video and languages in which video is available
- Comments, Duration and web-address of the video
- Related talks
- Description and Transcript of the video

METHODOLOGY



DATA ACQUISITION

The first step in the pipeline aims at importing our dataset into our environment. In our case; we will import the dataset containing information about 4005 unique TED talks. Each TED talk has 18 features to predict the views.

DATA AT A GLANCE

talk_id	4005	non-null	int64
title	4005	non-null	object
speaker_1	4005	non-null	object
all_speakers	4001	non-null	object
occupations	3483	non-null	object
about_speakers	3502	non-null	object
views	4005	non-null	int64
recorded_date	4004	non-null	object
published_date	4005	non-null	object
event	4005	non-null	object
native_lang	4005	non-null	object
available_lang	4005	non-null	object
comments	3350	non-null	float64
duration	4005	non-null	int64
topics	4005	non-null	object
related_talks	4005	non-null	object
url	4005	non-null	object
description	4005	non-null	object
transcript	4005	non-null	object

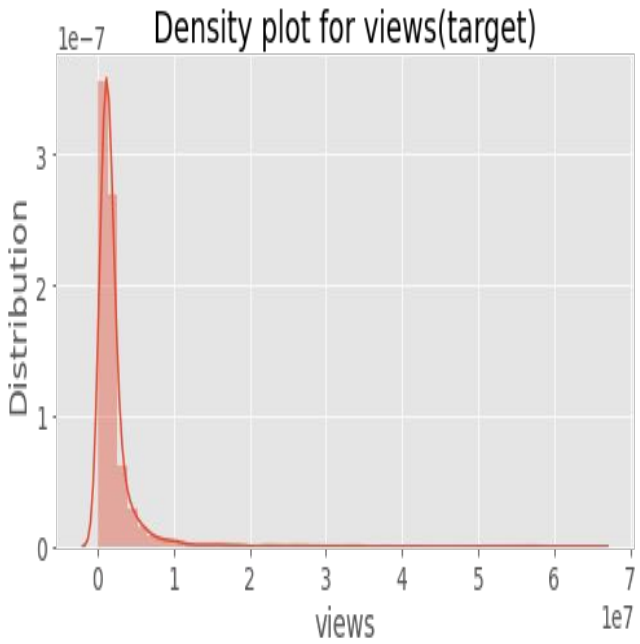
EXPLORATORY DATA ANALYSIS

1. We observe that we have 4005 entries in our dataset and some of the columns contain null values.
2. It is checked that there are no duplicate rows in our dataset which means that we have data on 4005 unique TED talks.
3. Columns containing information about speakers and languages in form of dictionaries and lists are input into the dataset as strings.
4. ‘Transcript’ and ‘Description’ contain large amount of textual data.
5. The following table describes the numerical columns in the dataset.

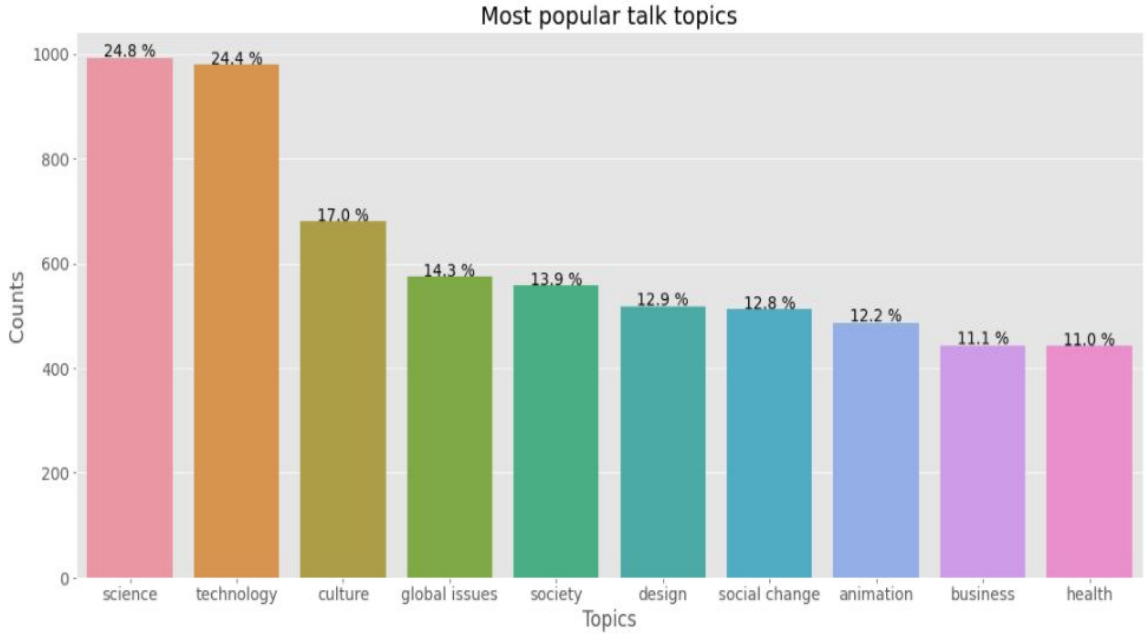
	count	mean	std	min	25%	50%	75%	max
talk_id	4005.0	1.243254e+04	1.744758e+04	1.0	1252.0	2333.0	23777.0	62794.0
views	4005.0	2.148006e+06	3.451226e+06	0.0	882069.0	1375508.0	2133110.0	65051954.0
comments	3350.0	1.619970e+02	2.688389e+02	0.0	38.0	89.0	188.0	6449.0
duration	4005.0	7.240112e+02	3.617755e+02	60.0	393.0	738.0	974.0	3922.0

UNIVARIATE ANALYSIS

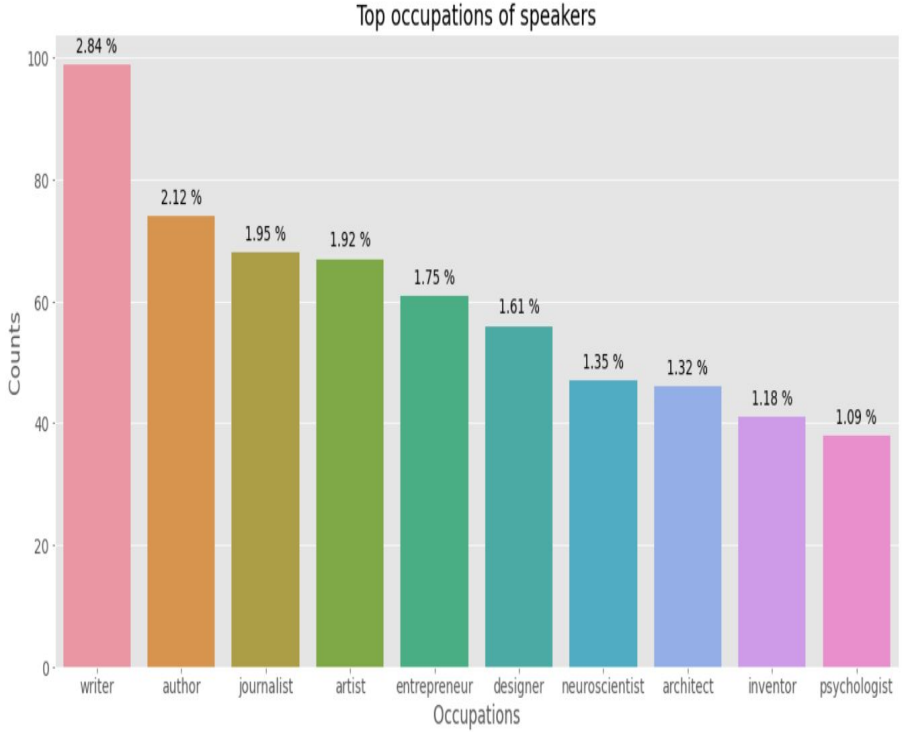
Following are the important observations we obtained from Univariate Analysis

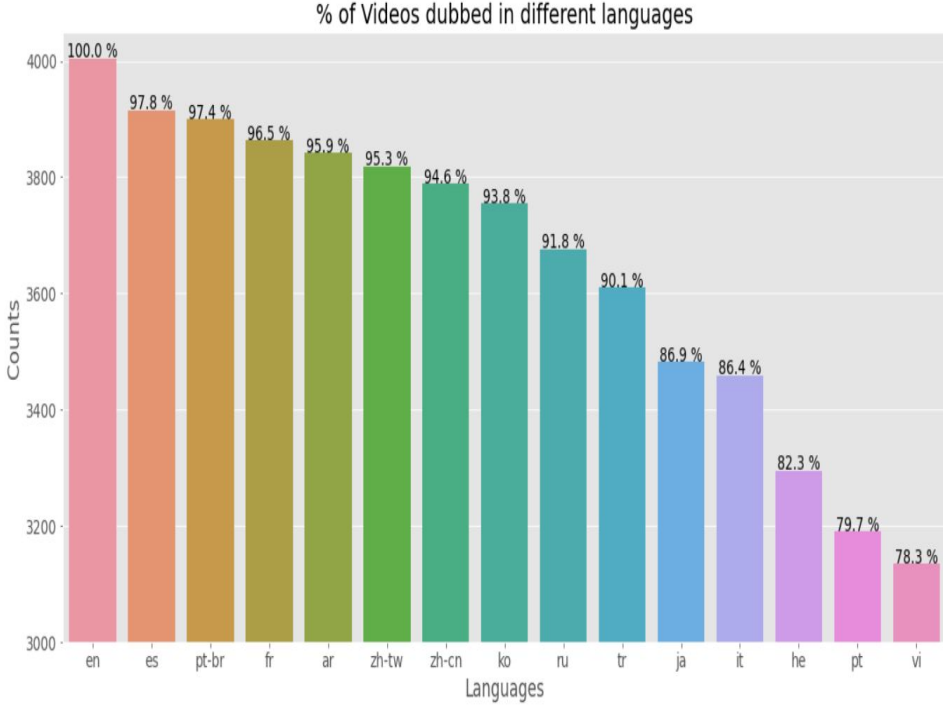


Distribution of ‘Views’ of TED talks



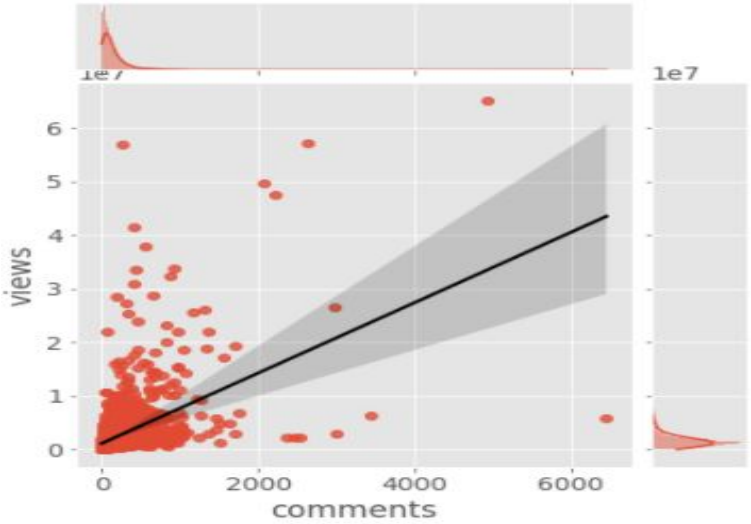
Plot showing Topics To Feature Most Number Of Times


Top occupations of speakers
2.84 %
2.12 %
1.95 %
1.92 %
1.75 %
1.61 %
1.35 %
1.32 %
1.18 %
1.09 %
Counts
Occupations
writer
author
journalist
artist
entrepreneur
designer
neuroscientist
architect
inventor
psychologist

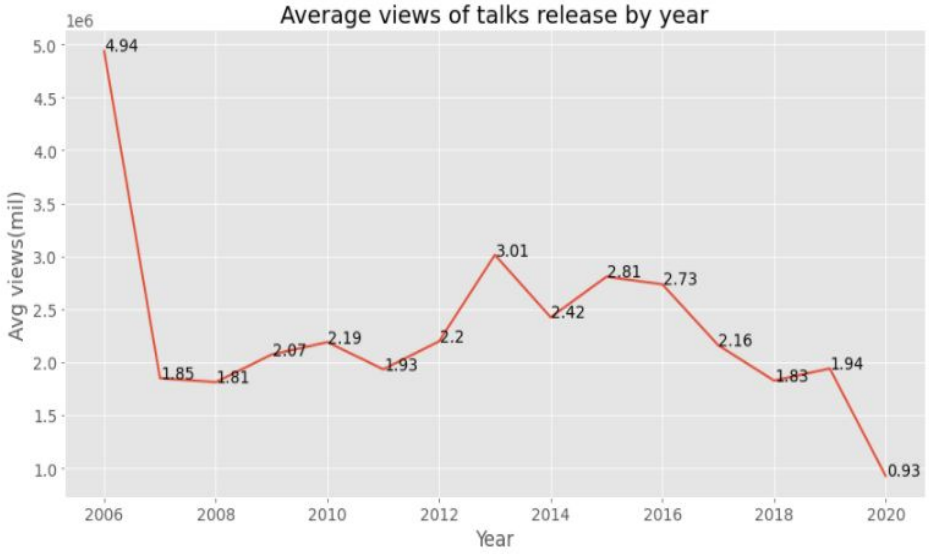

% of Videos dubbed in different languages
100.0 %
97.8 %
97.4 %
96.5 %
95.9 %
95.3 %
94.6 %
93.8 %
91.8 %
90.1 %
86.9 %
86.4 %
82.3 %
79.7 %
78.3 %
Counts
Languages
en
es
pt-br
fr
ar
zh-tw
zh-cn
ko
ru
tr
ja
it
he
pt
vi

BIVARIATE ANALYSIS

Following are the important observations from Bivariate Analysis.

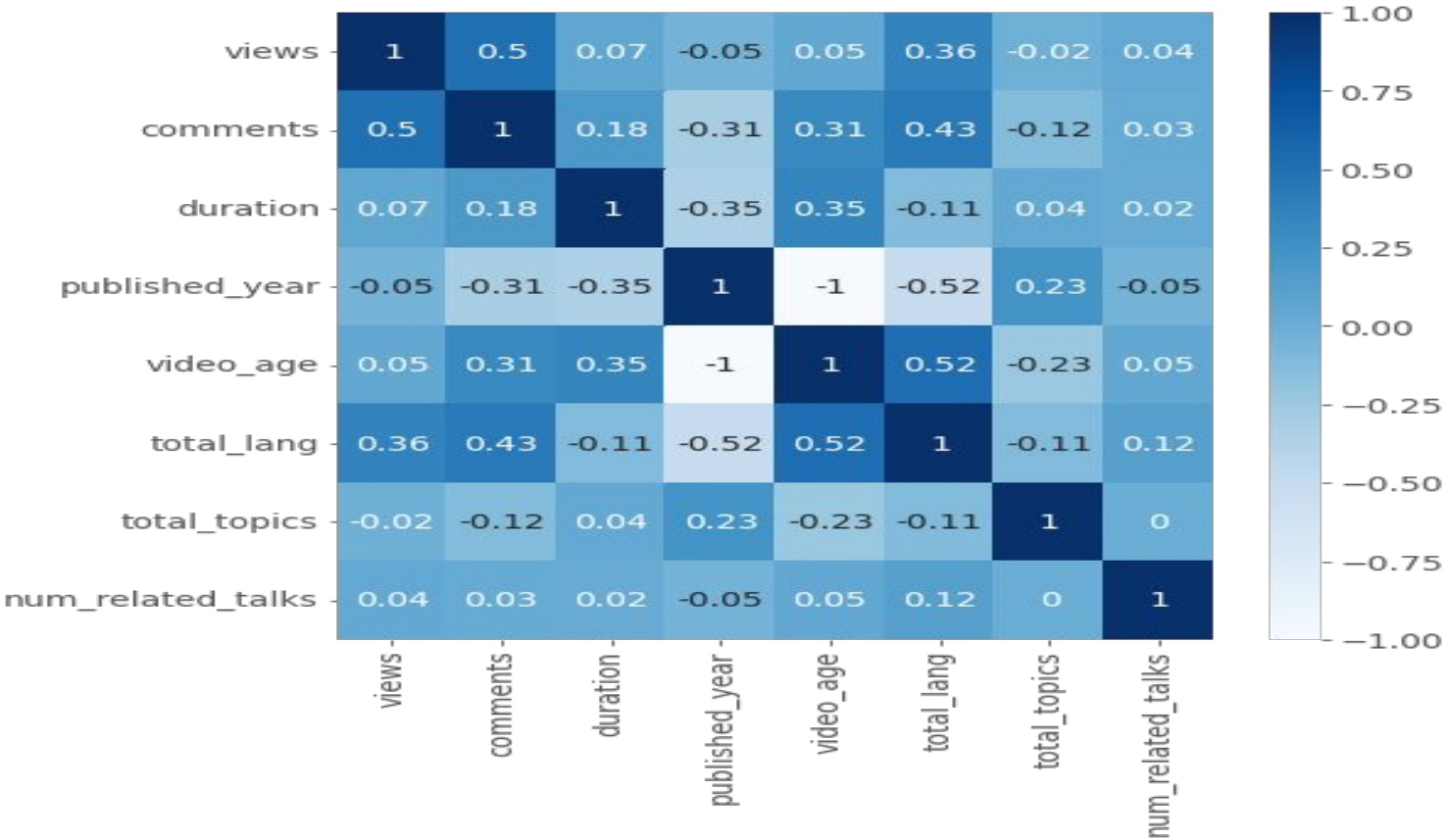


Plot of Comments Vs Views for various TED talks



Variation of Avg. Views By Published Year

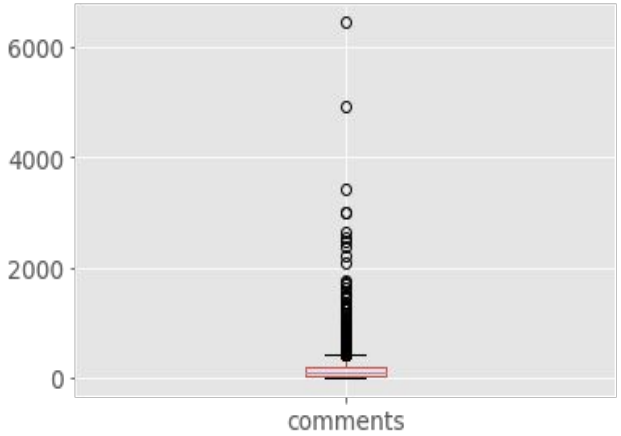
MULTIVARIATE ANALYSIS

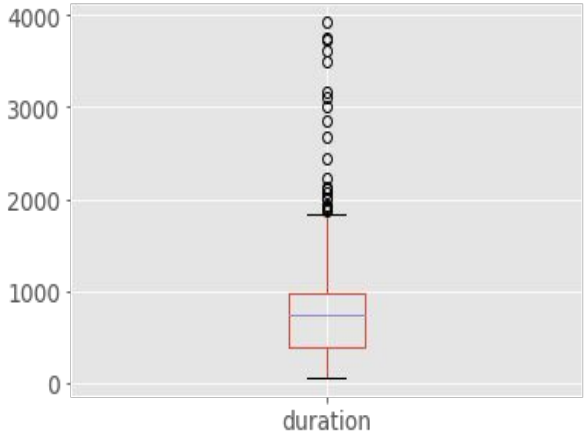


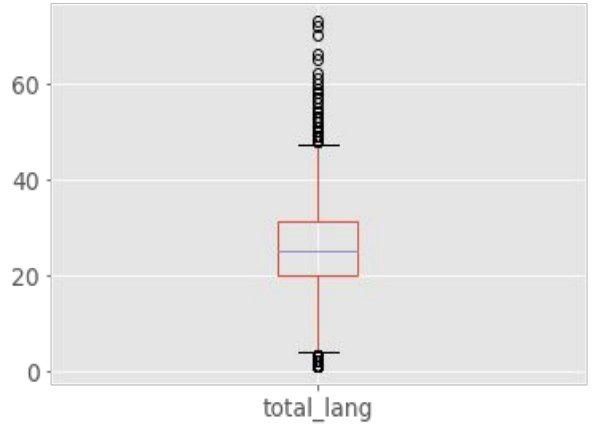
HEATMAP SHOWING CO-RELATION AMONG NUMERICAL FEATURES

DATA CLEANING

On analyzing the numerical features in our dataset; we found outliers in most of the features.







TRAIN TEST SPLIT

After cleaning our data; the dataset is split into Train - Test datasets. This is done to ensure that our test dataset is completely isolated and there is no information leakage during the training process of machine learning models.

DATA PREPROCESSING AND FEATURE ENGINEERING

In this stage, we are creating two types of features:

1) Numerical Features
2) Numerical Word2Vector embedded feature vectors from corpus

For numerical features:

- Features like all_speakers, occupations, about_speakers are first filled with with a value as ‘others’ where Nan was present. After that, these features are converted to a dictionary representation from a string of dictionary representation.
- Features like published_date and recorded_date are converted to datetime.
- Total_days_since_published is also created using published_date and recored_date.

- New features like day, month, year and week_day are created using datetime objects created in previous step.
- More features like speaker_1_average_views, topic_wise_average_views, unique_topics and event_average_views are introduced in the data by grouping data according to speaker_1, topics, and event respectively.

For numerical word2Vector embedded features:

1. Used Google's pre-trained word2vec model.
2. Created a corpus using transcript feature.
3. Cleaned the corpus by removing stop-words, digits, punctuation marks, etc.
4. After cleaning the corpus, feataure_vectors are created with dimension as 300.

DATA MODELLING

Many models were trained, from simple parametric models like Linear Regression to tree based models. It was observed that linear regression did not performed up to the mark and tree based models generally outperformed them. The gist of top performing regression models is given.

LGBM Regressor - It is a boosting technique that uses tree based learning algorithm. It grows tree leaf wise rather than level wise.

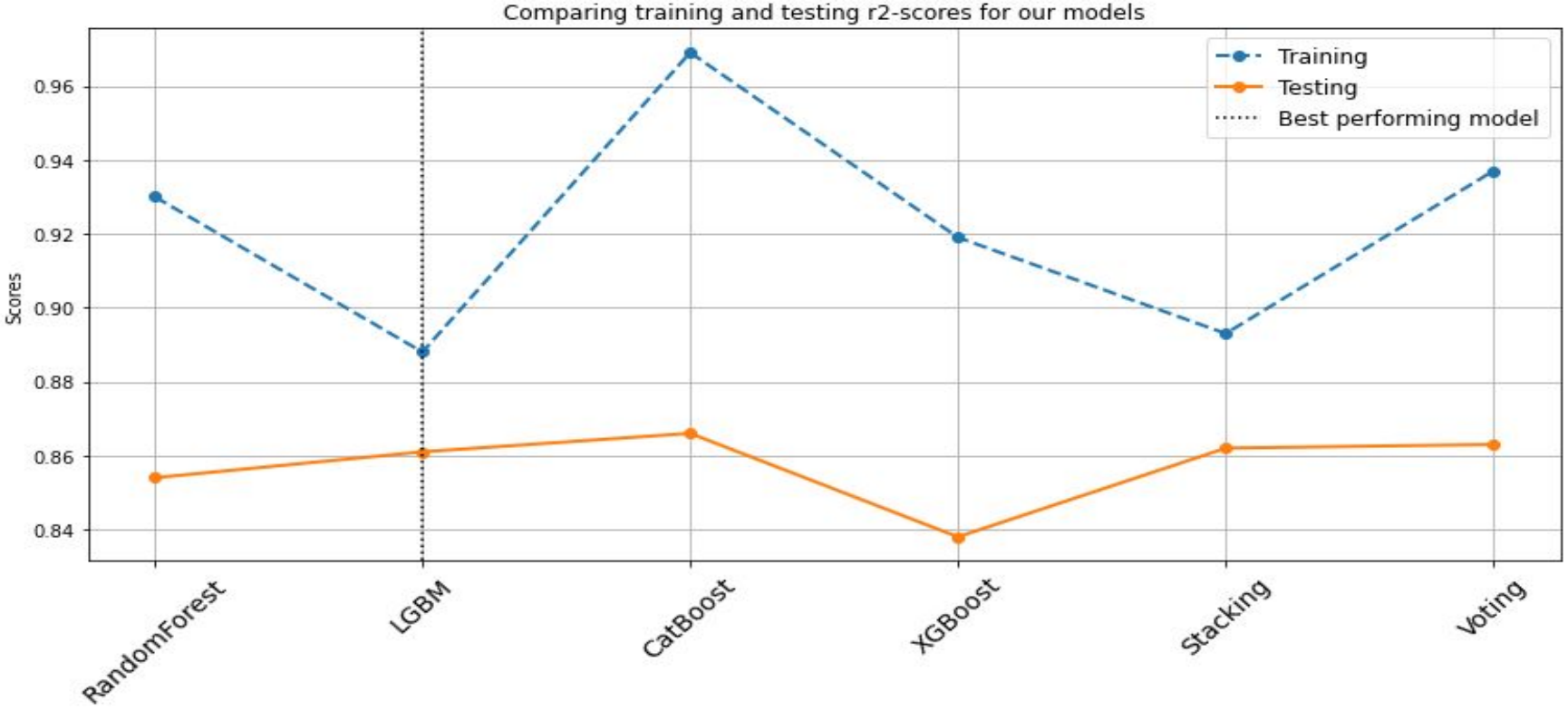
XGBoost Regressor - It is also a boosting technique that uses gradient descent algorithm to minimize the loss when adding new tree models.

Stacked Regressor - CatBoost, LGBM, RandomForest and XGBoost were chosen as the base models for the Stacked Regressor and LGBM as the final one. It tried to combine them in such a way that scores increase.

Voting Regressor - The estimators used for voting were the same ones being used in Stacking.

RESULTS OF DATA MODELLING

MODEL	TRAIN-R2	TEST-R2
LGBM	0.888	0.861
XGBoost	0.919	0.838
Stacked Model	0.893	0.862
Voting Model	0.937	0.863

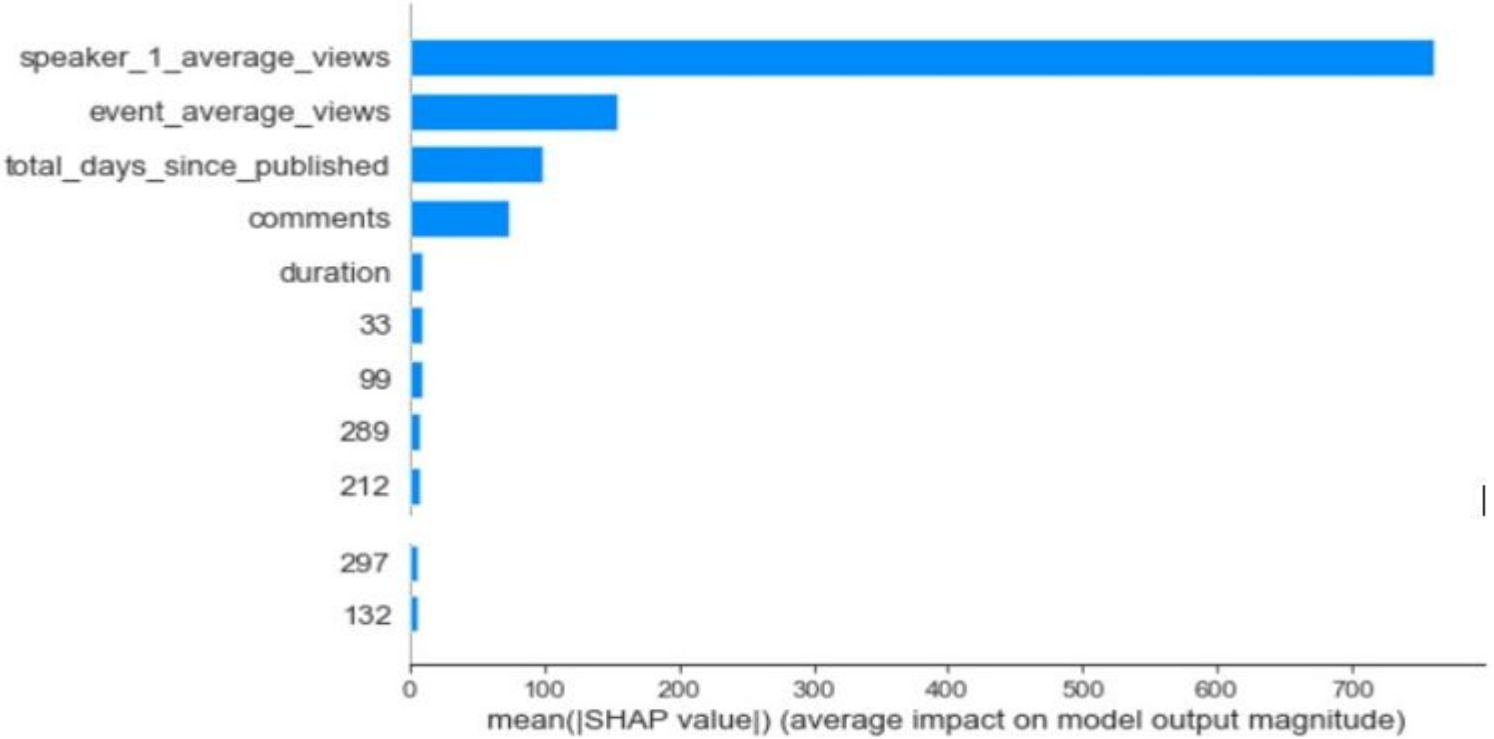


TRAIN AND TEST INTERPRETATION

LGBM is chosen as the final model for our regression problem owing to best test result and close Train and Test R2-scores.

MODEL INTERPRETATION Using SHAP

SHAP values interpret the impact of having a certain value for a given feature in comparison to the prediction we'd make if that feature took some baseline value.



CONCLUSION

- The main objective was to build a predictive model, which could help in predicting the views of the videos uploaded on the TEDx website.
- We have built a model where it is able to predict what views next TEDx video would get. The model was able to predict with an **R2** score of **0.861** on the test data.
- Model Interpretation also shows how each feature contributes to the predicted views.
- We built a baseline model and then improved on that.
- We have done modelling using: LGBM, XGBOOST, stacked model, voting model, RandomForest and CATBOOST.
- We have used different error metrics like MAE, MSE and RMSE. The errors were minimum for LGBM.
- Hyper-parameter tuning helped us to get rid of overfitting.
- We interpreted the model using SHAP.

CHALLENGES FACED

- Creating a quality corpus was difficult here. We tried with multiple features like description, transcript, etc. for how many features should be taken.
- We tried TF-IDF, countvectorizer, and word2vec using gensim but weren't able to create a good corpus which was fitting nicely to data.
- Features were created using a mix of feature vectors from the corpus and numerical columns.
- We also tried topic modelling here but again it did not give good results.
- A lot of feature engineering was required.
- Because of a large number of features, we were facing some overfitting. So, reaching to an optimal model was challenging.

FUTURE SCOPE OF WORK

- We would want to build a quality corpus using most of the textual features here.
- Time features are available, we could also try time series modelling.
- Since the data has textual features and as sequence is important in text, we could also try a BiDirectional LSTM as it could give good results.
- Creating Application and Model Deployment.
- Various Other regressors can be used for this problem.