TED Talks Views Prediction

Project Summary :

Problem Statement :

The main objective is to build a predictive model, which could help in predicting the views of the videos uploaded on the TEDx website.

About the Data :

We have the data of previous TED talk events , which contains data points such as the length (duration) of the talk, topics , speaker occupation and textual features such as Transcript , Title , and Description And most importantly , the target variable : the view of the video The Data is available for 4005 TED talks .

Dataset info

- Number of records: 4,005
- Number of attributes: 19

Features information:

The dataset contains features like:

- **talk_id**: Talk identification number provided by TED
- **title**: Title of the talk
- **speaker_1**: First speaker in TED's speaker list
- **all_speakers**: Speakers in the talk
- **occupations**: Occupations of the speakers
- **about_speakers**: Blurb about each speaker
- **recorded_date**: Date the talk was recorded
- **published_date**: Date the talk was published to TED.com
- **event**: Event or medium in which the talk was given
- **native_lang**: Language the talk was given in
- **available_lang**: All available languages (lang_code) for a talk
- **comments**: Count of comments
- **duration**: Duration in seconds
- **topics**: Related tags or topics for the talk
- **related_talks**: Related talks (key='talk_id',value='title')

- **url**: URL of the talk
- **description**: Description of the talk
- **transcript**: Full transcript of the talk

Target Variable :

- **views**: Contains Count of views of every talk

Approach taken :

The task was divided into 2 main parts :

1. **Statistical Analysis** over the dataset to discover relationships between each feature and the target variable . So that this relationship information can be used by the management in making better Business decisions
2. **Creating a Machine Learning Pipeline** , that can take in the data of any new video and predict how many views it will generate on a daily basis .It was required to kepp this pipeline modular , such that it can be retrained often when new data is collected

Project Work flow

1. Importing Libraries
2. Loading the dataset
3. Data Cleaning
4. EDA on features
5. Feature selection
6. Fitting the regression models
7. HyperParameter Tuning
8. Evaluation Metrices of the model
9. Final selection of the model
10. Conclusion

Technical Details for ML :

We used many Algorithms (Random Forest , XGBoost and CatBoost) We used **RandomSearchCV** for HyperParameter Tuning Comparing both R2 Score , we can see that Random Forrest and XGBoost model performs the best

Technical Insights from exploring the Data :

● For the ML Pipeline , the XGBoost Model performed the best ● For the NLP Pipeline , the Random Forest Model performed the Best ● Feature Engineering and Feature Extraction helped in increasing the model performance

Conclusions : Insights from exploring the Data :

● Topics like Technology , Science , Education , Biology attract the attention of viewers more than other topics . ● Entrepreneurs and Activists are the most engaging speakers

Python Libraries used

Datawrangling :

- Numpy
- Pandas

For Graphing :

- Matplotib
- Seaborn

Machine learning :

- Scikit-Learn
- SK-Opt
- XGBoost
- CatBoost

Miscellaneous :

- Google colab tools

Contributing Member:

Name	Email
Bharath P	Bharath0924.bp@gmail.com