INTERNSHIPSTUDIO

Project : Youtube adview Prediction

Data Description

The file train.csv contains metrics and other details of about 15000 youtube videos. The metrics include number of views, likes, dislikes, comments and apart from that published date, duration and category are also included.

The train.csv file also contains the metric number of adviews which is our target variable for prediction.

Data :

Use the below link to download the Data Set:
https://drive.google.com/file/d/1Dv-HF10AUUA03AO_cQvar462eXawk0iQ/view?usp=sharing

Context :

Youtube advertisers pay content creators based on adviews and clicks for the goods and services being marketed. They want to estimate the adview based on other metrics like comments, likes etc. The problem statement is therefore to train various regression models and choose the best one to predict the number of adviews. The data needs to be refined and cleaned before feeding in the algorithms for better results.

Attribute Information

'vidid' : Unique Identification ID for each video

'adview' : The number of adviews for each video 'views' : The number of unique views for each video

'likes' : The number of likes for each video
'dislikes' : The number of likes for each video

'comment' : The number of unique comments for each video

'published' : The data of uploading the video

'duration' : The duration of the video (in min. and seconds)

'category' : Category niche of each of the video

Objective

To build a machine learning regression to predict youtube adview count based on other youtube metrics.

Steps and Tasks

1. Import the datasets and libraries, check shape and datatype.
2. Visualise the dataset using plotting using heatmaps and plots. You can study data distributions for each attribute as well.
3. Clean the dataset by removing missing values and other things.
4. Transform attributes into numerical values and other necessary transformations
5. Normalise your data and split the data into training, validation and test set in the appropriate ratio.
6. Use linear regression, Support Vector Regressor for training and get errors.
7. Use Decision Tree Regressor and Random Forest Regressors.
8. Build an artificial neural network and train it with different layers and hyperparameters. Experiment a little. Use keras.
9. Pick the best model based on error as well as generalisation.
10. Save your model and predict on the test set.