Predictive Modeling for Live TV Viewership Using Machine Learning Techniques

This repository contains code and data files for the project " Predictive Modeling for Live TV Viewership Using Machine Learning Techniques." The project focuses on accurately predicting TV viewership metrics based on time series data using Random Forest Regressor, XGBoost and LightGBM.

Acknowledgements

This project is a collaborative effort by the following individuals:

- Yilin Yang
- Ediz Alatli (Admongrel)
- Prof. Alastair Moore (Project Supervisor)

Their contributions and expertise have been invaluable in the development and success of this project.

Contents

The repository contains all the code files for this project, as follows:

Archive
It contains techniques and models that were explored in this project but were ultimately not adopted. They are uploaded here for documentation and reference purposes.

Feature extraction_PyFlux_NowTV.ipynb: This Jupyter Notebook attempts to use the PyFlux ARIMA (Autoregression Integrated Moving Average) model for feature extraction on Now channel. However, it was not executed due to the high computational demands. Instead, this project uses Prophet for feature extraction.

LSTM_NowTV.ipynb: This Jupyter Notebook uses an LSTM model to implement multi-task output and predict future 24-hour viewership by applying a rolling window. However, this technique was proven to be time-inefficient under current computational conditions, and therefore, this model was not adopted.

01 Data Preprocess
This section contains the steps for pre-processing the raw data of the four channels. It collates and transforms the data based on temporal granularity.

Transform dataframe_NowTV.ipynb: This Jupyter Notebook details the preprocessing steps for Now data.

Transform dataframe_Kanal_D.ipynb: This Jupyter Notebook details the preprocessing steps for Kanal D data.

Transform dataframe_Star_TV.ipynb: This Jupyter Notebook details the preprocessing steps for Star TV data.

Transform dataframe_TV8.ipynb: This Jupyter Notebook details the preprocessing steps for TV8 data.

02 Exploratory Data Analysis
This section includes Exploratory Data Analysis (EDA) for data from four channels. Each code file within this section covers outliers, trend and seasonality, and correlation.

EDA_NowTV.ipynb: This Jupyter Notebook details the EDA steps for Now data.

EDA_Kanal_D.ipynb: This Jupyter Notebook details the EDA steps for Kanal D data.

EDA_Star_TV.ipynb: This Jupyter Notebook details the EDA steps for Star TV data.

EDA_TV8.ipynb: This Jupyter Notebook details the EDA steps for TV8 data.

03 Feature Extraction and Feature Selection
This section includes the detailed steps of feature extraction using Prophet for the four channel data. In particular, the steps for feature selection using Mutual Information and XGBoost for extracted features are also included in the code file of Now.

Feature_selection_Prophet_H_NowTV.ipynb: This Jupyter Notebook contains feature extraction and feature selection steps for Now data.

Feature selection_Kanal_D.ipynb: This Jupyter Notebook contains feature extraction step for Kanal D data.

Feature selection_Star_TV.ipynb: This Jupyter Notebook contains feature extraction step for Star TV data.

Feature selection_TV8.ipynb: This Jupyter Notebook contains feature extraction step for TV8 data.

04 Model Training
This section uses Now channel data to train three models: Random Forest Regressor, XGBoost, and LightGBM. Specifically, each model employs a MultiOutput Regressor to achieve multivariable output. For each model, the performance is tested using three different rolling window sizes.

RandomForest_rw_add_NowTV.ipynb: This Jupyter Notebook trains and tests the performance of three different window sizes using the Random Forest on Now data.

XGBoost_rw_add_NowTV.ipynb: This Jupyter Notebook trains and tests the performance of three different window sizes using the XGBoost on Now data.

LightGBM_rw_add_NowTV.ipynb: This Jupyter Notebook trains and tests the performance of three different window sizes using the LightGBM on Now data.

05 Model Evaluation
This section retains the best-performing window size from the training of the three models on Now data for hyperparameter tuning and final evaluation. Specifically, the process involves

further tuning of the models using Now data, saving the best models, and then testing the final models on data from all four channels.

Model_Evaluation_NowTV.ipynb: This Jupyter Notebook includes hyperparameter tuning and final evaluation using Now data.

Model_Evaluation_Kanal_D.ipynb: This Jupyter Notebook includes model applicability test using Kanal D data.

Model_Evaluation_Star_TV.ipynb: This Jupyter Notebook includes model applicability test using Star TV data.

Model_Evaluation_TV8.ipynb: This Jupyter Notebook includes model applicability test using TV8 data.

Note

The data used in this project and the stored best models can be obtained from the Google Drive repository. The raw data for the four channels (used in the 01 Data Preprocess section) were filtered from the full data provided by Admongrel. The data used in subsequent code files can be obtained and stored by following the steps in this repository's code files (modify file paths as needed during execution).

Some code files in this repository were run using Google Colab, accessing the database through Google Drive. When running these code files locally, you can remove the code linking to Google Drive, modify the file paths, and use local data instead.

For any further clarification, please contact yilin.yang.23@ucl.ac.uk.