News Popularity Prediction on Facebook

Yue Zhuang
Brown University DSI
Git: `https://github.com/zysophia/News_Rank_Prediction`

1 Introduction

1.1 Problem Description

In this project, we dig into the popularity of news on the Facebook News platform. With the timing of publishment an important feature that affects the popularity of news, we will develop a real time predicting system for the popularity of certain piece of news.

The target variable of our project would be the popularity rate in the near future (20 minutes) of a certain piece of news.

The problem is regression since we will predict the popularity rate of news, which is a continuous variable.

The prediction of popularity rate of news can be of significant importance for the recommend systems. Social media platforms will be able to push the most popular piece of news to consumers by introducing the real-time prediction model to their existing recommend system.

1.2 Dataset Description

1.2.1 Data Source

Our data comes from the UCI dataset: `https://archive.ics.uci.edu/ml/datasets/News+Popularity+in+Multiple+Social+Media+Platforms`

1.2.2 Dataset Structures

In this section, we will describe the dataset – News Popularity in Multiple Social Media Platforms Data Set. This is a large data set of news items and their respective social feedback on multiple platforms. The collected data relates to a period of 8 months, between November 2015 and July 2016, accounting for about 100,000 news items on four different topics: economy, microsoft, obama and palestine.

- **news final sheet**

Column Name	**Type**	**Brief Description**
IDLink	numerical	Unique identifier of news items
Title	string	Title of the news item according to the official media sources
HeadLine	string	Headline of the news item according to the official media sources
Source	categorical	Original news outlet that published the news item
Topic	categorical	Query topic used to obtain the items in the official media sources
Publishdate	numerical	Date and time of the news items' publication
SentimentTitle	numerical	Sentiment score of the text in the news items' title
SentimentHeadline	numerical	Sentiment score of the text in the news items' headline
FaceBook	numerical	Final value of the popularity according to Facebook
GooglePlus	numerical	Final value of the popularity according to Google+
LinkedIn	numerical	Final value of the popularity according to LinkedIn

Figure 1: News Final Sheet

In this csv file, each row represents a certain piece of news, with total 93240 rows.

- **times series sheet**

Column Name	**Type**	**Brief Description**
IDLink	numerical	Unique identifier of news items
TS1	numerical	Level of popularity in time slice 1 (0-20 minutes upon publication)
TS2	numerical	Level of popularity in time slice 2 (20-40 minutes upon publication)
...	...	...
TS144	numerical	Final level of popularity after 2 days upon publication

Figure 2: Time Series Sheet

In these csv files, each file includes data for the time series popularity rate for news on a certain social media platform and with certain topics.

2 EDA

In this section, we perform exploratory data analysis and show the results that we found the most informative.

2.1 Popularity v.s. Topics

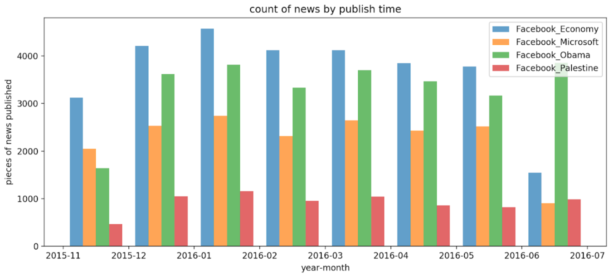


Figure 3: Count of news by publish time

Above is a barplot of the number of news published on Facebook during November 2015 and July 2016, grouped by its respective topic. The data seems quite balanced here.

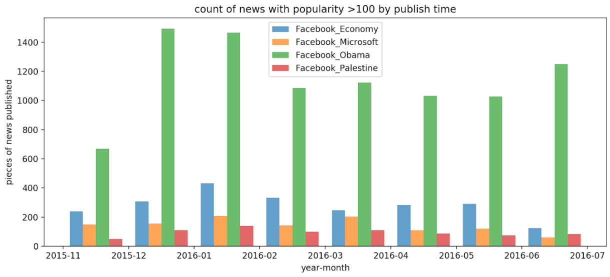


Figure 4: Count of popular news by publish time

Above is the same barplot but focuses on only those news with a total popularity rate greater than 100. We may find out from the plot that topics related with Obama is quite popular during 2015 and 2016, which indicates that we are having an unbalanced data set.

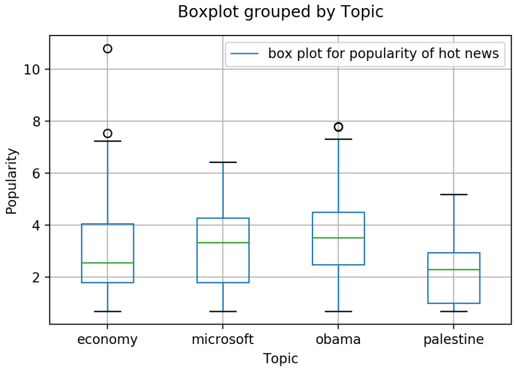


Figure 5: Box plot for popularity of hot news

The boxplot shows similar results about the unbalance characteristic.

2.2 Popularity Trend

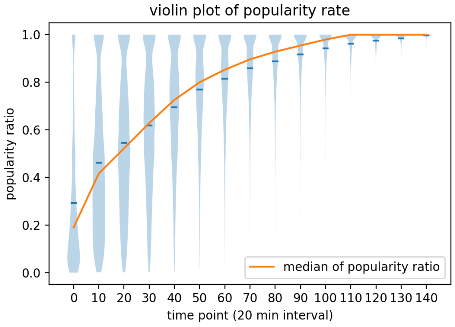
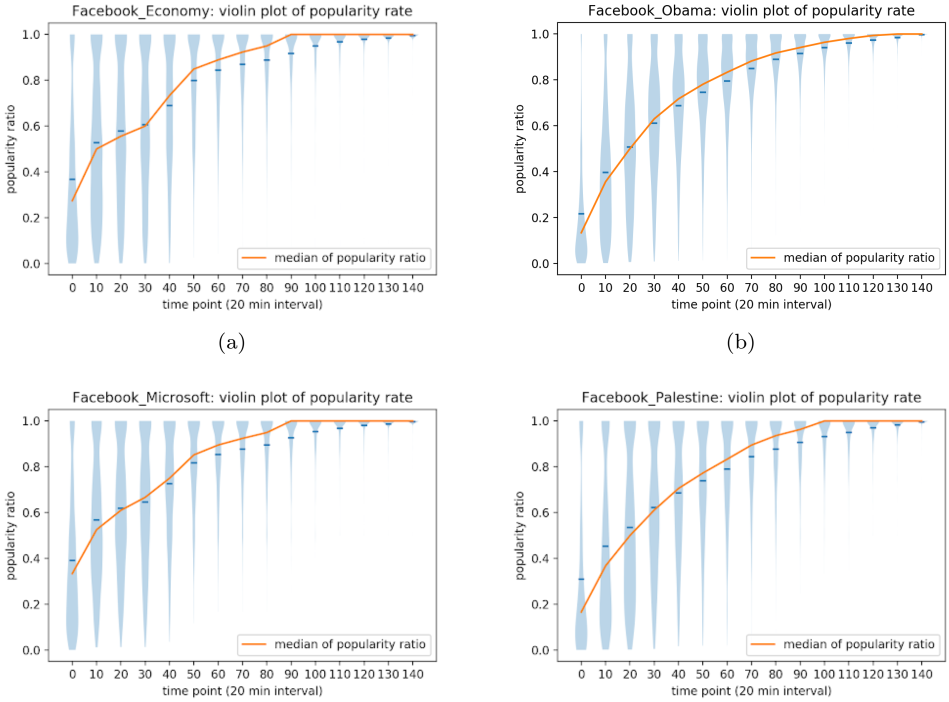


Figure 6: Violin plot of popularity rate

The violin plot gives the trend of popularity rate ratio for a piece of news during the first two days after its publishment. We could observe that the piece of news reaches an average 30% of its total popularity in the first 20 minutes after publishment, consistent with the intuition that the news is most popular when it is first published while quickly lose popularity as time goes on.



(a) (b)

(c) (d)

Figure 7: Violin Plot of Popularity Trend By Topics

The violin plots above shows the popularity trend for news with different topics. We could observe that the popularity trend of news is indeed closely related to its respective topic, where the plotting of Economy and Microsoft is bimodal while that of Obama and Palestine is quite smooth.

2.3 News Popularity V.S. Topic Popularity

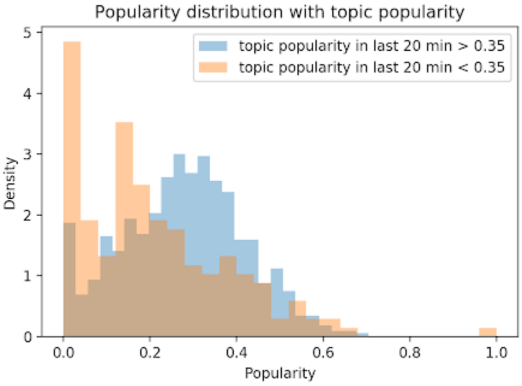


Figure 8: Histogram of Popularity Distribution by Topic Popularity (20m)

We can observe that when the corresponding topic is not popular, the news is very likely to be unpopular. However, the most popular news always appears where the topic is not popular.

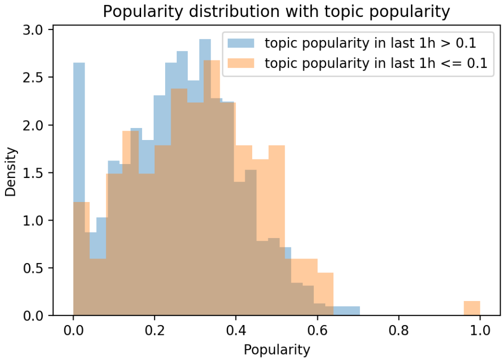


Figure 9: Histogram of Popularity Distribution by Topic Popularity (1h)

To dig into the embedding reasoning for the scenario, we looked back into a longer period (1 hour) to check. The figure above shows that when we look back into a longer period, the most popular news still lies where the topic is not popular, but the news is not necessarily likely to be unpopular when the topic is not popular. We may make a bold conjecture that facebook is holding an imperfect recommendation system, and the news has not yet been exposed to users when the topic is not hot enough.

3 Methods

3.1 Data Preprocessing

3.1.1 Missing Data

Before merging the five raw dataframes, we would like to check the missing values. We have a missing rate of approximately 0.3% and the MCAR test gives a result of 1.0. Thus we drop the rows with missing values.

3.1.2 Basic Calculation

To get more informative features for the model, we conducted some basic calculations for the change rate of popularity and the corresponding topic popularity, and then merge the dataframes.

The resulting dataframe has 37 columns.

3.1.3 Encoder and Scaler

- **One Hot Encoder:** We applied the one hot encoder on the categorical data, topics.
- **MinMax Scaler:** We applied minmax scalers on all the continuous features excluding the sentiment value which is already normalized.
- **Logarithm:** We applied logarithm on the continuous features before scaling since we have extreme values.

3.1.4 Final DataFrame

The merged dataframe is a 121049 * 40 matrix.

Additionally, as is mentioned by Nuno Moniz and Luis Torgo [2], when making predictions, we shall actually focus more on those data with a higher popularity rate. Thus, we have selected those rows with a target popularity greater than e05 for our model. The final dataframe is a 3412 * 40 matrix.

3.2 ML Methods Overview

3.2.1 ML models and parameters

We will first split the data into training (80%) and testing (20%), and then we will use kfold to conduct grid search for the best parameters. We have set the number of folds to be 5 here. Also, we have randomly set the random states for train_test splitting for 10 times, going from 42 to 420 stepped by 42. We will measure the uncertainty caused by data splitting later.

3.2.2 Evaluation Metrics

We will use mean square error on the cross validation data to search for the best parameters since we are dealing with a regression problem. And then we calculate both the mean square error and the R square score on the test data to better evaluate the model outcome.

3.3 Random Forest Regressor

3.3.1 Random Forest Regressor Parameters

- **Max_depth:** Each integer in the range [6,10].
- **Min_samples_split:** Min_samples_split: In [0.025, 0.05, 0.1, 0.2].

3.3.2 Random Forest Regressor Uncertainty

To eliminate the bias caused by the selection of random state of the random forest model, we conducted an inner loop through the random_state in [42, 84, 126], and then take the average as the outcome of the model.

To calculate the uncertainty due to the setting of random state, after we have selected the best parameters, we shall go through 10 random states to calculate the standard deviation of MSE and R2 score.

3.4 Gradient Boosting Regressor

3.4.1 Gradient Boosting Regressor Parameters

- **Max_depth:** Each integer in the range [3,8].
- **Min_samples_split:** Min_samples_split: In [0.025, 0.05, 0.1, 0.2].

3.4.2 Gradient Boosting Regressor Uncertainty

The procedure is the same as that of random forest regressor.

3.5 SVR

3.5.1 SVR Parameters

- **C:** In [1, 10, 100, 1000].
- **gamma:** In [0.01, 0.1, 1.0, 10.0].

4 Results

4.1 Baseline Scores

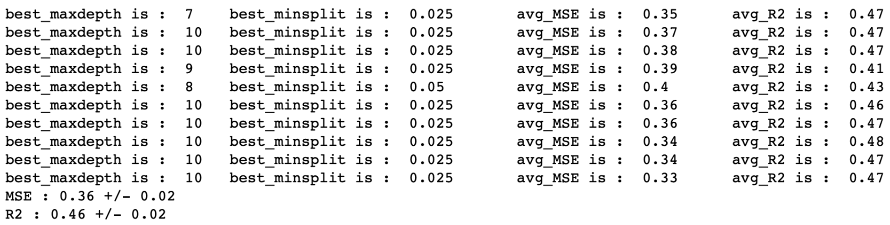
We calculate the R2 score of the models to compare with baseline models. The baseline score of R2 is 0, indicating that the model explains none of the variability of the response data around its mean.

4.2 Parameters, Scores and Feature Importances

In this section, we will discuss the best parameters of the models, the scores and feature importances.

4.2.1 Random Forest Regressor

- **Parameter selection**



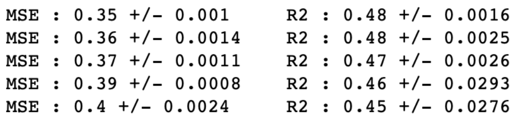
```
best_maxdepth is :  7    best_minsplit is :  0.025      avg_MSE is :  0.35      avg_R2 is :  0.47
best_maxdepth is :  10   best_minsplit is :  0.025      avg_MSE is :  0.37      avg_R2 is :  0.47
best_maxdepth is :  10   best_minsplit is :  0.025      avg_MSE is :  0.38      avg_R2 is :  0.47
best_maxdepth is :  9    best_minsplit is :  0.025      avg_MSE is :  0.39      avg_R2 is :  0.41
best_maxdepth is :  8    best_minsplit is :  0.05       avg_MSE is :  0.4       avg_R2 is :  0.43
best_maxdepth is :  10   best_minsplit is :  0.025      avg_MSE is :  0.36      avg_R2 is :  0.46
best_maxdepth is :  10   best_minsplit is :  0.025      avg_MSE is :  0.36      avg_R2 is :  0.47
best_maxdepth is :  10   best_minsplit is :  0.025      avg_MSE is :  0.34      avg_R2 is :  0.48
best_maxdepth is :  10   best_minsplit is :  0.025      avg_MSE is :  0.34      avg_R2 is :  0.47
best_maxdepth is :  10   best_minsplit is :  0.025      avg_MSE is :  0.33      avg_R2 is :  0.47
MSE : 0.36 +/- 0.02
R2 : 0.46 +/- 0.02
```

Figure 10: RF Parameter GridSearch

We may observe that a max_depth of 10 and a min_samples_split of 0.025 is pretty reasonable for the model.

- **Uncertainty by splitting**
  We have an average MSE of 0.36 (with standard deviation 0.02) and an average R2 score of 0.46 (with standard deviation 0.02, greater than baseline). The uncertainty comes from data splitting.

- **Uncertainty by random state**
  Fix the parameters to be the best ones (max_depth = 10, min_samples_split = 0.025). Run the model 5 times to estimate the uncertainty by random state.



```
MSE : 0.35 +/- 0.001      R2 : 0.48 +/- 0.0016
MSE : 0.36 +/- 0.0014     R2 : 0.48 +/- 0.0025
MSE : 0.37 +/- 0.0011     R2 : 0.47 +/- 0.0026
MSE : 0.39 +/- 0.0008     R2 : 0.46 +/- 0.0293
MSE : 0.4 +/- 0.0024      R2 : 0.45 +/- 0.0276
```

Figure 11: RF Random State Uncertainty

The uncertainty caused by the random state of the model is actually pretty low.

- **Feature importances**

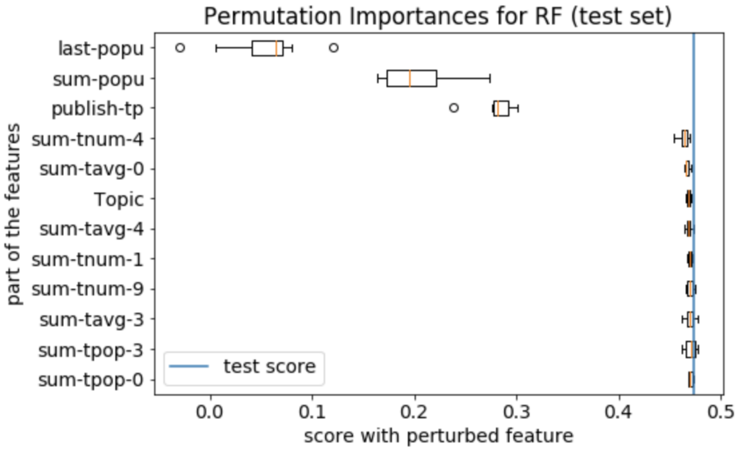
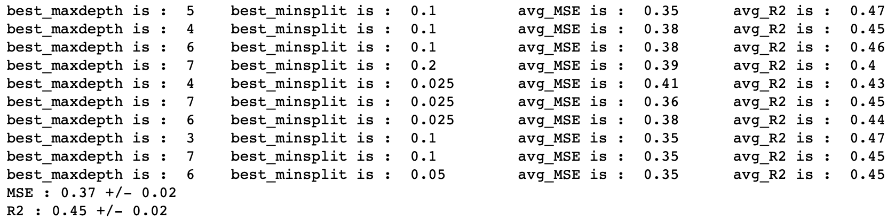


Figure 12: RF Permutation Feature Importances

The figure above shows the permutaion importances for the most important features among all. We could observe that the popularity history and publish time of the news is important for prediction.

4.2.2 Gradient Boosting Regressor

- **Parameter selection**



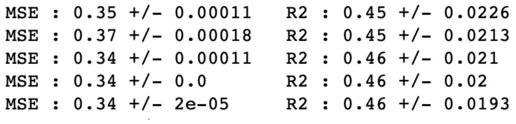
```
best_maxdepth is :  5    best_minsplit is :  0.1        avg_MSE is :  0.35      avg_R2 is :  0.47
best_maxdepth is :  4    best_minsplit is :  0.1        avg_MSE is :  0.38      avg_R2 is :  0.45
best_maxdepth is :  6    best_minsplit is :  0.1        avg_MSE is :  0.38      avg_R2 is :  0.46
best_maxdepth is :  7    best_minsplit is :  0.2        avg_MSE is :  0.39      avg_R2 is :  0.4
best_maxdepth is :  4    best_minsplit is :  0.025      avg_MSE is :  0.41      avg_R2 is :  0.43
best_maxdepth is :  7    best_minsplit is :  0.025      avg_MSE is :  0.36      avg_R2 is :  0.45
best_maxdepth is :  6    best_minsplit is :  0.025      avg_MSE is :  0.38      avg_R2 is :  0.44
best_maxdepth is :  3    best_minsplit is :  0.1        avg_MSE is :  0.35      avg_R2 is :  0.47
best_maxdepth is :  7    best_minsplit is :  0.1        avg_MSE is :  0.35      avg_R2 is :  0.45
best_maxdepth is :  6    best_minsplit is :  0.05       avg_MSE is :  0.35      avg_R2 is :  0.45
MSE : 0.37 +/- 0.02
R2 : 0.45 +/- 0.02
```

Figure 13: GBR Parameter GridSearch

We may observe that a max_depth of 6 and a min_samples_split of 0.1 is pretty reasonable for the model.

- **Uncertainty by splitting**
  We have an average MSE of 0.37 (with standard deviation 0.02) and an average R2 score of 0.45 (with standard deviation 0.02, greater than baseline).

- **Uncertainty by random state**
  Fix the parameters to be the best ones (max_depth = 6, min_samples_split = 0.1). And then run the model 5 times to estimate the uncertainty by random state of the gradient boosting model.



```
MSE : 0.35 +/- 0.00011   R2 : 0.45 +/- 0.0226
MSE : 0.37 +/- 0.00018   R2 : 0.45 +/- 0.0213
MSE : 0.34 +/- 0.00011   R2 : 0.46 +/- 0.021
MSE : 0.34 +/- 0.0       R2 : 0.46 +/- 0.02
MSE : 0.34 +/- 2e-05     R2 : 0.46 +/- 0.0193
```

Figure 14: GBR Random State Uncertainty

The uncertainty caused by the random state of the model is actually pretty low.

- **Feature importances**

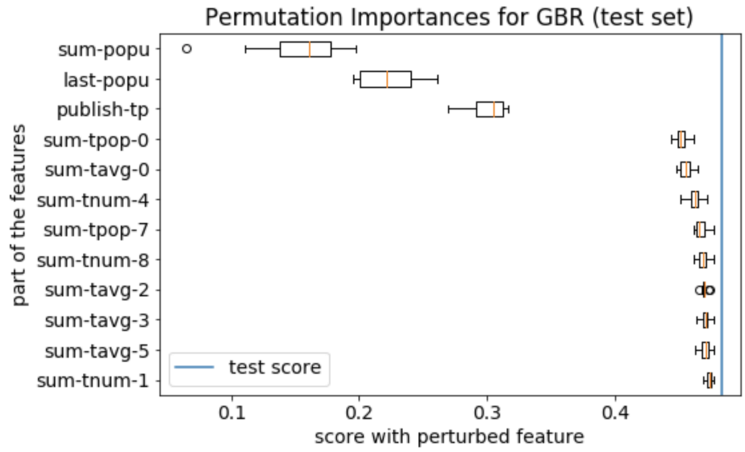
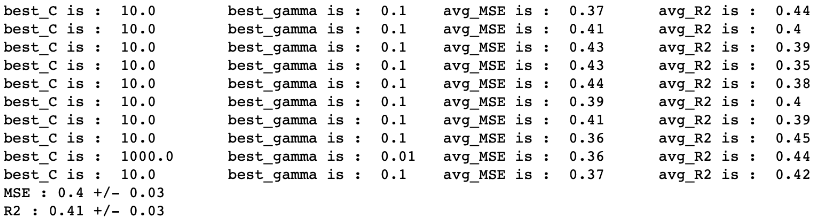


Figure 15: GBR Permutation Feature Importances

We could observe that the popularity history, topic polularity and publish time of the news is important for prediction.

4.2.3 SVR

- **Parameter selection**



```
best_C is :  10.0       best_gamma is :  0.1     avg_MSE is :  0.37      avg_R2 is :  0.44
best_C is :  10.0       best_gamma is :  0.1     avg_MSE is :  0.41      avg_R2 is :  0.4
best_C is :  10.0       best_gamma is :  0.1     avg_MSE is :  0.43      avg_R2 is :  0.39
best_C is :  10.0       best_gamma is :  0.1     avg_MSE is :  0.43      avg_R2 is :  0.35
best_C is :  10.0       best_gamma is :  0.1     avg_MSE is :  0.44      avg_R2 is :  0.38
best_C is :  10.0       best_gamma is :  0.1     avg_MSE is :  0.39      avg_R2 is :  0.4
best_C is :  10.0       best_gamma is :  0.1     avg_MSE is :  0.41      avg_R2 is :  0.39
best_C is :  10.0       best_gamma is :  0.1     avg_MSE is :  0.36      avg_R2 is :  0.45
best_C is :  1000.0     best_gamma is :  0.01    avg_MSE is :  0.36      avg_R2 is :  0.44
best_C is :  10.0       best_gamma is :  0.1     avg_MSE is :  0.37      avg_R2 is :  0.42
MSE : 0.4 +/- 0.03
R2 : 0.41 +/- 0.03
```

Figure 16: SVR Parameter GridSearch

We may observe that a C of 10 and a gamma of 0.1 is pretty reasonable for the model.

- **Uncertainty by splitting**
  We have an average MSE of 0.4 (with standard deviation 0.03) and an average R2 score of 0.41 (with standard deviation 0.03, greater than baseline). The uncertainty comes from data splitting.

- **Feature importances**

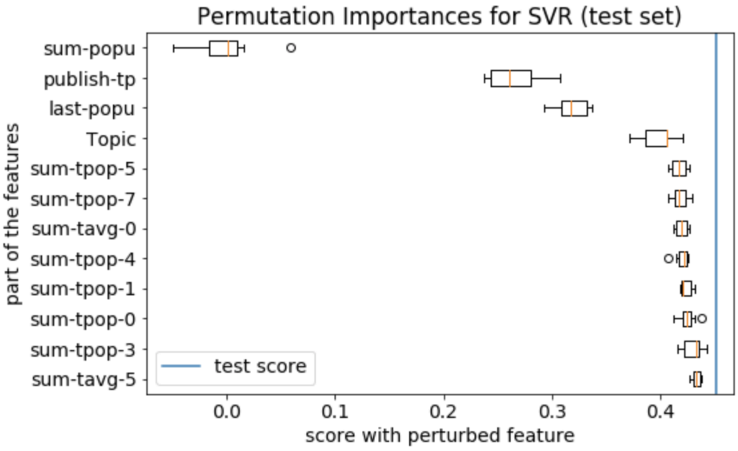


Figure 17: SVR Permutation Feature Importances

We could observe that the popularity history, publish time and topic of the news is important for prediction.

4.3 Model Comparison

The outcome of the random forest model is the best of the three, with lower MSE and higher R2 score.

4.4 Business and Academic Interpretations

As is shown in the model outcomes, we could obtain a reasonably low MSE with a comparatively high R2 score performing either the ensenble models or the svr model. The random forest regressor has the best performance amang all three.

The uncertainty of the prediction caused by either data splitting or non-deterministic models is comparatively low, indicating that we will be able to get a stable prediction performance.

The permutation importances of features appears to be different under three models. However, by observing the three feature importance measures, we could draw the conclusion that the future popularity rate of a piece of news is closely related to its popularity history and publish time. The topic popularity and the corresponding topic itself as well contribute to the prediction. This observation agrees with our intuition when dealing with news popularity.

5 Outlook

The model gives a reasonable outcome but there is still space for improvement.

Our model has 36 features, while a lot of them are closely correlated with each other. To maintain the interpretability of the model, we did not perform dimension reduction in our project. We may try to reduce some columns so as to improve the efficiency of training.

The sentiment features appear to be less informative in our prediction, which is unexpected. We may have performed some pre-calculation for the sentiment features to detect people's attitude toward a topic under certain periods. We may also collect data for the author of the news as extra features.

The tree based model performs better in our predictions than the support vector one, we may try other methods like neural network to see if it can get even better.

References

[1] Nuno Moniz, Luis Torgo. *Multi Source Social Feedback of Online News Feeds.* 2018

[2] Nuno Moniz, Luis Torgo. *The Utility Problem of Web Content Popularity Prediction.* 2018

[3] Friedman, J. H. *Greedy Function Approximation: A Gradient Boosting Machine.* 1999

[4] Cortes, Corinna, Vapnik, Vladimir N. *Support-vector networks.* 1995

[5] Ho, Tin Kam. *Random Decision Forests.* 1995

[6] UCI Data Source. *https://archive.ics.uci.edu/ml/datasets/News+Popularity+in+Multiple+Social+Media+Platforms*. 2018