Kaggle Competition:Predicting Online News Popularity

Team 3-NN: Kseniya Bout, Balint Van, Leonardo Byon

ABSTRACT

Although we tried many things, we could not improve significantly our benchmark randomforest prediction.

INTRODUCTION

We were provided a dataset of online news articles published by Mashable.com with the goal of developing a multi-class classifiers for predicting the level of popularity for each article. The dataset is composed by 60 features and consists of 39,000 articles, of which 30,000 are labeled training data, with the remainder serving as a test set. The popularity is a function of the number of times the article has been shared, where total number of classes is five, where (1) are those that were shared a few times, while (5) being those that went viral. The evaluation critiera is Classification Accuracy.

The current report describes the approaches taken by our team, 3-NN, to tackle the competition. The report is organized as follows: In Section 1, we present our exploratory analysis, followed by a description of our efforts to create new features. To conclude, in Section 3, we present our prediction models and results.

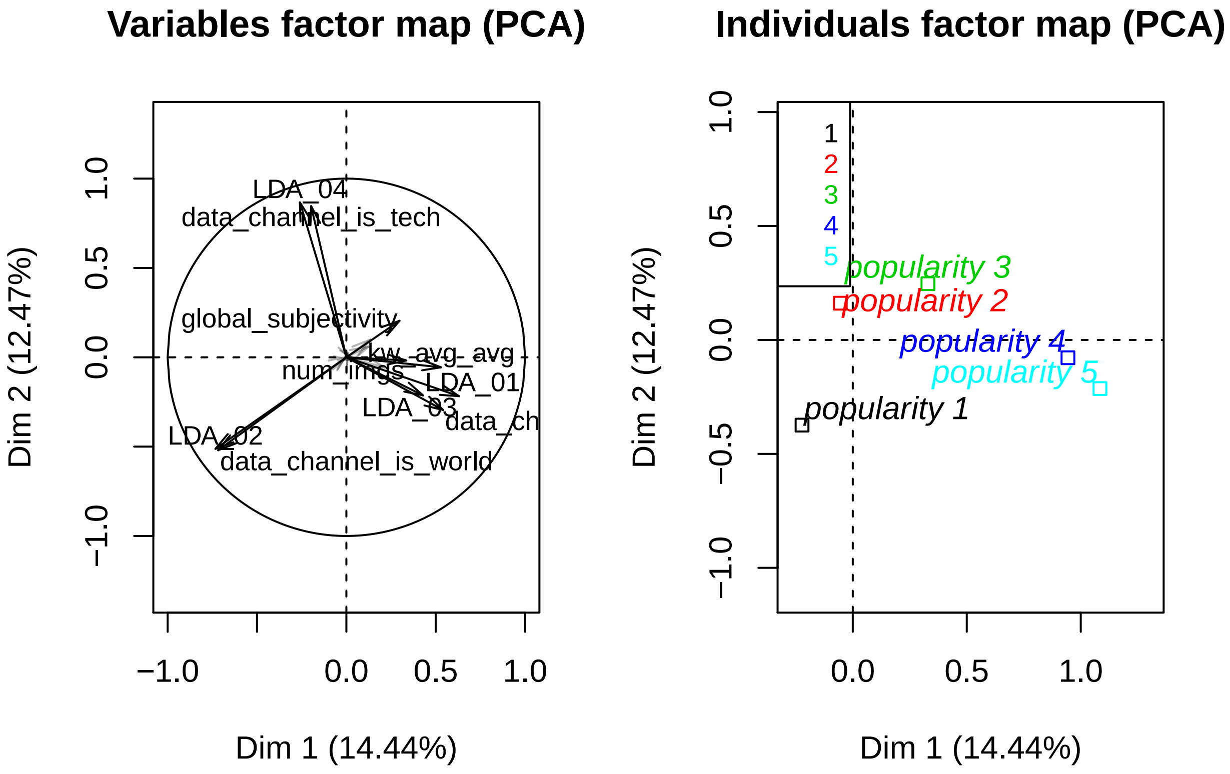
1. EXPLORATORY ANALYSIS

Outlier detection

To get an initial understanding of the data we looked at boxplots of the variables conditioned on popularity. One variable (id = 22686) has unreasonable values for the variables n_non_stop_words, n_unique_tokens and n_non_stop_unique_tokens. For all other observations the values are between 0 and 1, while for this observation they are over 500 for each of the three. We removed this observation from the training set.

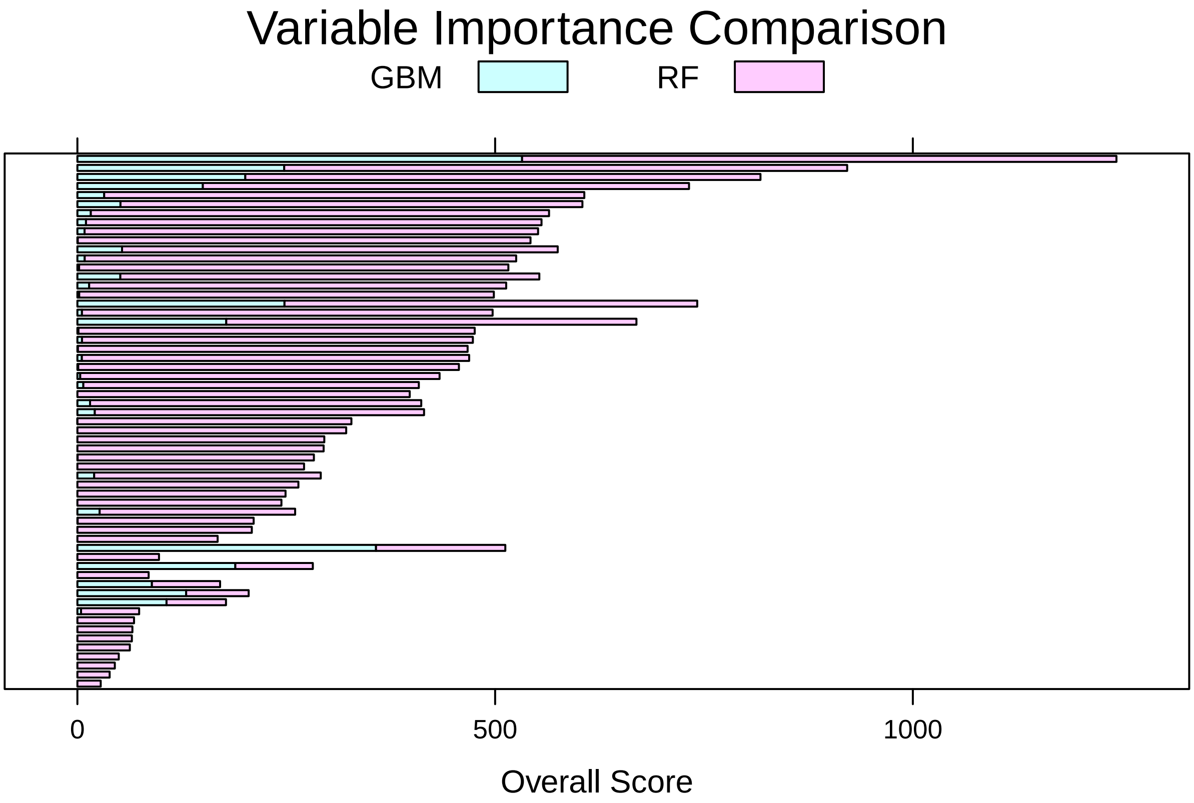
Principal Component Analysis (PCA)

PCA helped us gain better understanding on the relevance of features, where those closely associated with popularity were average number of key words and news category/topic. For example, there is positive correlation between popularity and the number of average key words and LDA_03 (undiscosed topic). Additionally, low popularity is related with World and Enterntainment news, while mid levels of popularity are associated with Technology-related news.



Variable Importance

To cross check our interpretations on variable importance of the PCA above, and to, additionally, understand how variable importance is determined by popular off-the-shelf ensemble methods, we implemented standard Random Forest (RF) and Gradient Boosting Machines (GBM). On one hand, RF decorrelated bagged trees by randomly selecting M predictors from the full set of P predictors for each tree at each iteration, where each tree is independent of the other trees, while on the other, GBM grows low-depth trees sequentially, by using error information obtained from previously grown trees. Results are displayed below, sorted by importance as per RF. Although significantly different, top features are similar with each method and consistent with PCA interpretations. We also notice that the regularization process of GBM is much more strict than that of RF, which suggests RF having higher potential overfitting risk.



Class Imbalance and Positive Skewness

In line with the expectations, the distribution of the popularity of online news article, follows a power law, creating high class imbalance is present in the training data, and can naturally expect the same distribution in the test data.

	Class 1	Class 2	Class 3	Class 4	Class 5
Freq	9478	13764	5712	999	47

In the current competition setup, the cost of Type I/II errors for any class is equal, and therefore we are not concerned with accurately prediction the low frequency of class (4) and (5) articles. Nevertheless, because accurately predicing them could provide a competitive edge, we tried accounting for class in our analysis. Possible strategies include allocating different weights across classes to increase sensitivity in favor of imbalanced classes, down/up sampling, synthetic minority over-sampling technique (SMOTE), and stratified sampling to equalize the number of obserations across classes. We tried up-sampling and stratified sampling with RF, but were unable to improve over our plain-vanilla RF benchmark.

Related to the class imbalance, we observe that many features presented significant positive skewness.

2. FEATURE ENGINEERING

To improve algorithmic performance, we applied logarithmic transformations, $log(1+x)$ to the abovementioned skewed features. Additionally, date-related information extracted from the article's URL was discretized (converting all factor levels into dummy variables).

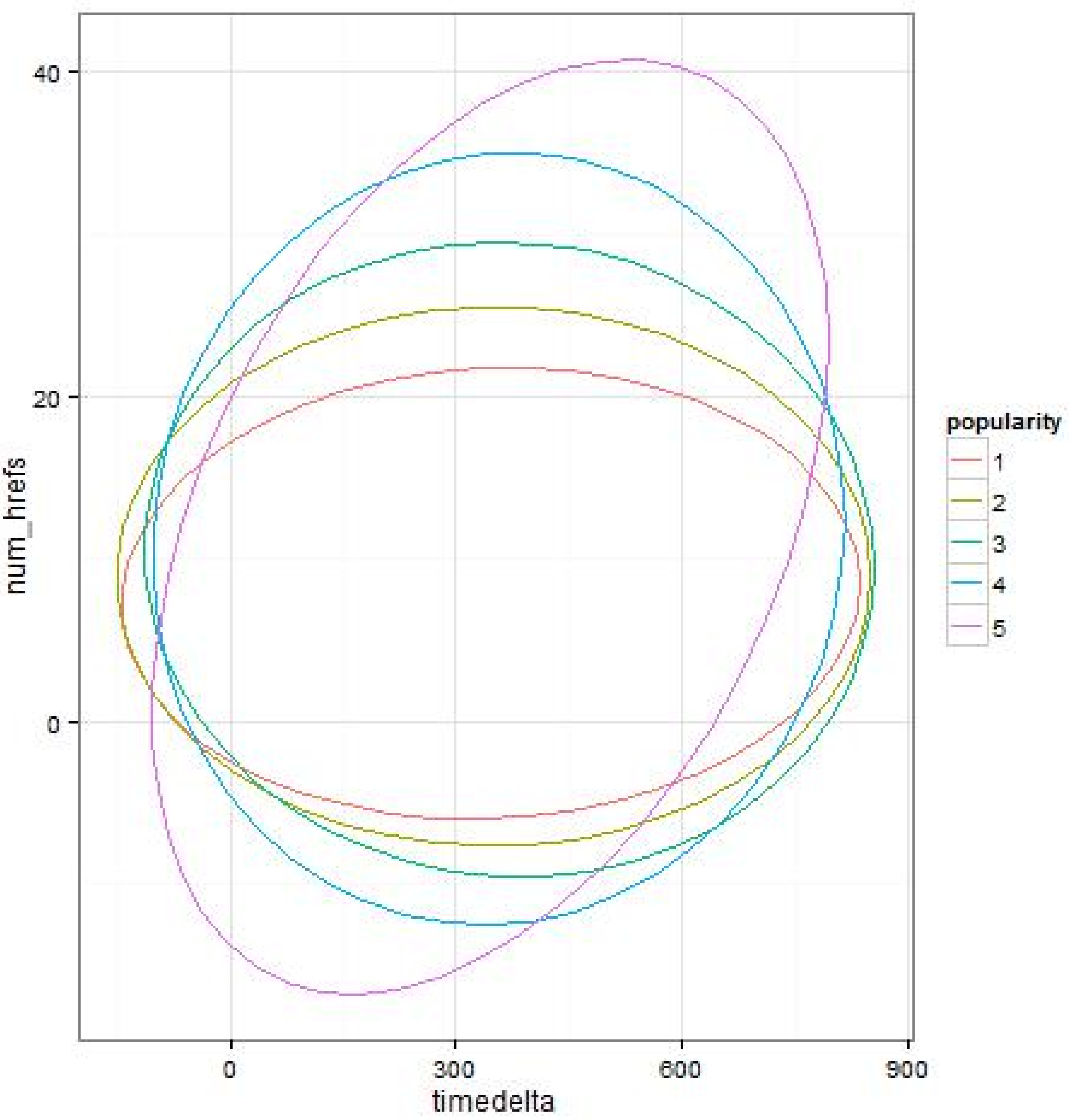
We extracted from the URL-s the words describing the topic of the article. After stemming, removing stopwords and selecting the most common words, we tried to include them in our model. The randomForest package could not handle well this many features, so we used the ranger package, which is more scalable. We also tried to use hierarchical clusters of the words as features instead of the words themselves. Neither of the above approches improved the accuracy of our prediction.

According to literature, category of the news, language subjectivity, news source, and frequency and relevance of key words (named entities), have been shown to be good predictors for online news popularity (Bandari et al, 2012). Among these four characteristics, all but the news source and relevance of keyword is provided in the original data set. In our context, we defined news source as the author of each article, which was acquired via web scraping. Out of the entire training and test set, we obtained 340 unique authors and 6200 NULL values, suggesting a high ratio of article to unique authors ratio. We ran standard GBM to identify the most important authors, of which only 37 proved to have predicting power. As a proxy for keyword relevance, we attempted to compute a relevance score using data obtained one week prior to each article's publishing date from Google Trend API, but quota limits were too restrictive to complete the queries.

Nevertheless, the addition of these new features did not help improve our standard RF benchmark.

Posteriorly, we added predictions made by K-NN, as new features for RF, but no improvement was observed.

We improved our model with the inclusion of the interaction feature timedelta * num_hrefs. We discovered this feature by looking at bivariate plots where each popularity cluster is represented by an ellipse in which 95% of the observations fall. We plotted these plots for variable pairs where one of the variables was among the most important predictors according to Random Forest. We were looking for plots where the ellipses do not overlap entirely and they are different along a diagonal, not a vertical or horizontal line.



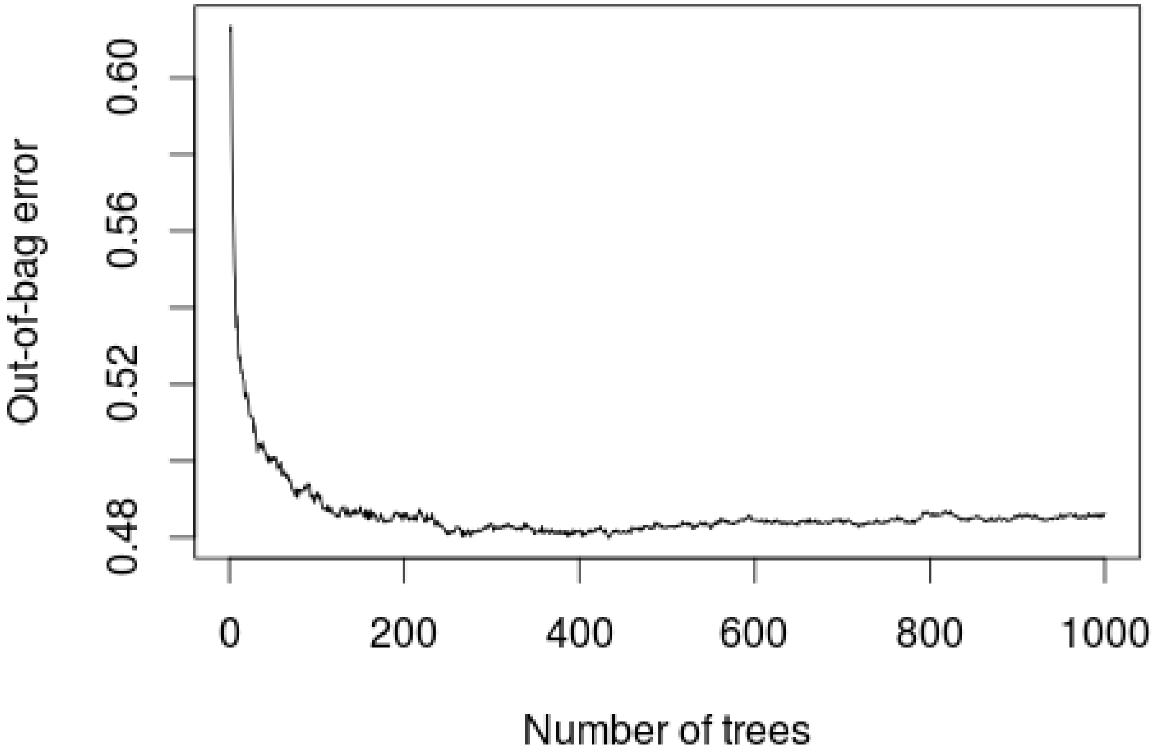
Encouraged by the surprising success of this methodology we have chosen based on these graphs 10 other interaction variables, but including them did not improve the accuracy.

3. PREDICTION MODELS AND RESULTS

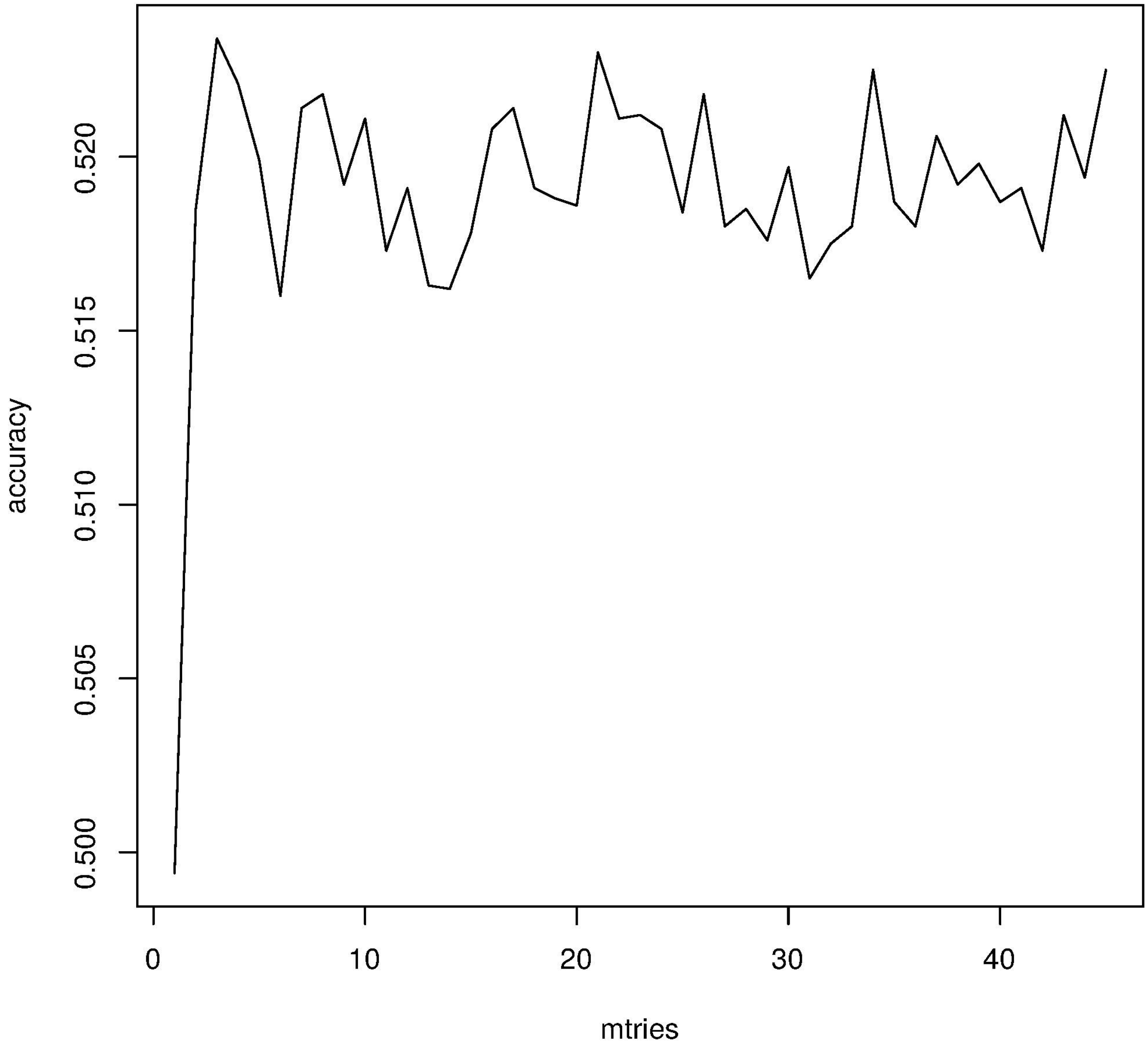
We used random forests as our prediction models, because Ren and Yang wrote that they are the best for this kind of data. We tried gradient boosting machines as well, but they did not perform better.

Parameter optimization

We tested different number of trees and we decided on ntree = 400, based on the following plot.



Number of variables randomly sampled as candidates at each split (mtry) does not have a clear effect on accuracy, as the variance is big. But to us there seemed to be a slight upward trend, so we chose a value (21) much bigger than the default (8).



Kaggle submissions

Here we describe the different submissions and their accuracy. We did not make a submission for all the methods we described above, as many times we could see without a submission that it would not be an improvement.

Dates	Description	Public_score	Private_score
07 Feb	Basic random forest model with 400 trees.	0.53571	0.52456
09 Feb	RF with wordcounts from the url.	0.50466	0.51222
11 Feb	RF with balanced training set.	0.53106	0.51856
12 Feb	Gradient boosting machine with dummified time variable.	0.52950	0.52156
14 Feb	RF with wordcluster features from url.	0.52950	0.52622

Dates	Description	Public_score	Private_score
15 Feb	Reproducible version of the basic model.	0.52329	0.51456
07 Mar	RF with variable selection and the author variable	0.38199	0.35178
12 Mar	RF with knn predictions as additional feature.	0.52640	0.52100
14 Mar	Fine tuned RF with an additional interaction term.	0.54503	0.52411

The best private score was achieved by a model which used the clustered words. But as it was faring worse on the public leaderboard and in our internal tests as well (did not test it many times due to long running time), we did not decide to work on it more. Our final submission was a slightly enhanced version of the initial RF, but in the private leaderboard it did a little bit worse than the initial submission. In any case, our "advanced" models did not work much better than our initial benchmark, the seed influenced our accuracy more than the decision among our best models.

BIBLIOGRAPHY

Bandari, R., Asur, S., Huberman, B.A.: The pulse of news in social media: Forecasting popularity. CoRR (2012)

Ren, He, and Yang, Quan.: Predicting and Evaluating the Popularity of Online News. (2015)