Online News Popularity

Ming-Yang Nian
Department of Statistics, National Chengchi University
March 04, 2021

Outline

PART 01/Introduction

Introduction

- With the rapid growth of online news services and social media, it is very beneficial if we could determine readers’ unseen behavioral patterns.
- We intend to make use of a large and recently collected dataset with over 39000 articles from Mashable website.
- For the purpose of research, various machine learning algorithms were applied to first select informative features and then analyze and compare the performance of several machine learning algorithms.

Motivation

In 2017, Online advertising officially surpassed TV to become the largest media, with NT $ 25.87 billion, first surpassing 22.53 billion for TV (including cable and TV) advertising.

To increase advertisement incomes, we need improve the popularity of articles .

We can set the advertisements which are related to our article topic, like sports, games, and so on.

Purpose

We want to figure out that if an **article will be popular or not(judge By median).**

-Relation between popularity of articles and variables. (Feature Selection)

-Build a model to predict the popularity of articles .

PART 02/Literature Review

Literature Review

- Predicting and evaluating the popularity of online news have been studied extensively in numerous papers.
- Ren et al. applied many machine learning algorithms like Linear Regression, Logistic Regression, Support Vector Machine, Random Forests , which the best accuracy can achieve 69%.
- Frenandes et al. used Random Forests, AdaBoost, Support Vector Machine which the best accuracy can achieve 66%.

PART 03/Analysis

Dataset

Online News Popularity

Number of variables: 61

Number of objects: 39644

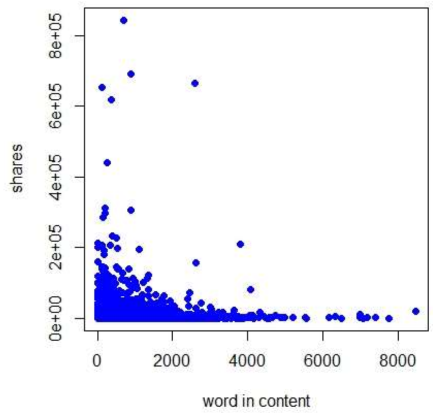
Published by the Mashable blog, recording articles published by the blog in 2 years

Feature	Type (#)
Words	
Number of words in the title	number (1)
Number of words in the article	number (1)
Average word length	number (1)
Rate of non-stop words	ratio (1)
Rate of unique words	ratio (1)
Rate of unique non-stop words	ratio (1)
Links	
Number of links	number (1)
Number of Mashable article links	number (1)
Minimum, average and maximum number of shares of Mashable links	number (3)
Digital Media	
Number of images	number (1)
Number of videos	number (1)
Time	
Day of the week	nominal (1)
Published on a weekend?	bool (1)

Feature	Type (#)
Keywords	
Number of keywords	number (1)
Worst keyword (min./avg./max. shares)	number (3)
Average keyword (min./avg./max. shares)	number (3)
Best keyword (min./avg./max. shares)	number (3)
Article category (Mashable data channel)	nominal (1)
Natural Language Processing	
Closeness to top 5 LDA topics	ratio (5)
Title subjectivity	ratio (1)
Article text subjectivity score and its absolute difference to 0.5	ratio (2)
Title sentiment polarity	ratio (1)
Rate of positive and negative words	ratio (2)
Pos. words rate among non-neutral words	ratio (1)
Neg. words rate among non-neutral words	ratio (1)
Polarity of positive words (min./avg./max.)	ratio (3)
Polarity of negative words (min./avg./max.)	ratio (3)
Article text polarity score and its absolute difference to 0.5	ratio (2)

Target	Type (#)
Number of article Mashable shares	number (1)

Preprocessing



- Cleaning: The dataset had been cleaned before it was uploaded to Kaggle.

- Preparing: We may find some trends about some variables like number of article words. We can delete metadata.

- Scaling: Scaling before we build the model can reduce the range influence among variables.

Feature Selection

- Correlation Filter Method
  Ex: rate of unique words in the content &
  rate of non-stop unique words in the content

- Purging superfluous variables
  Ex: published on which day &
  published on weekday or weekend

Build Models

Supervised learning

- Logistic Regression
- Decision Tree
- KNN
- SVM

Ensemble methods

- Random Forests
- Bagging
- Adaboost

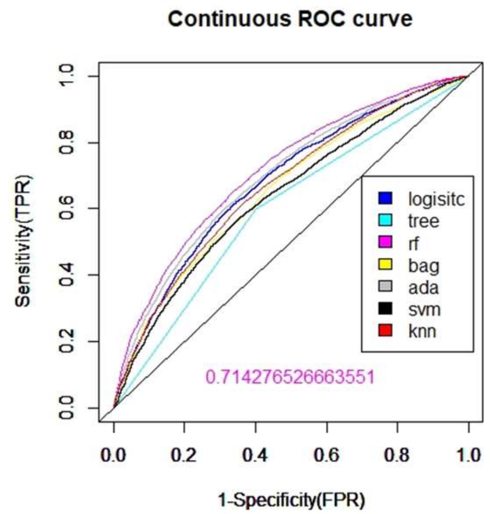
PART 04/Evaluation

Index of Evaluation

- Accuracy
- AUC(Area Under Curve)

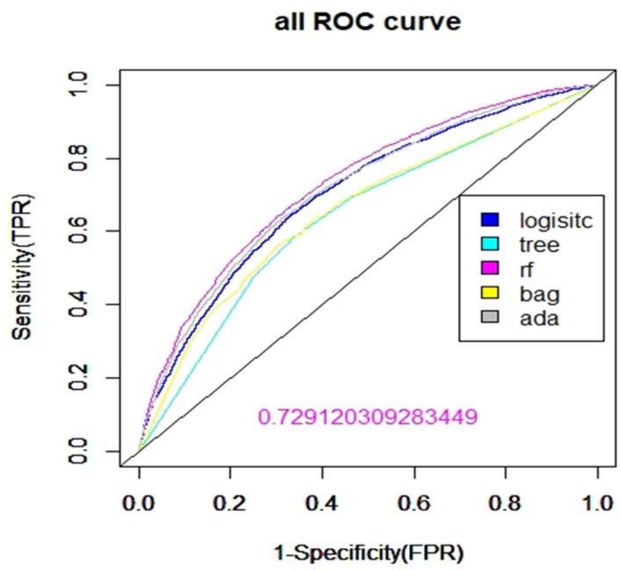
Performance under the continuous variables models

Algorithms	Accuracy(1 - True Error Rate)	AUC
Logistic Regression	0.636	0.680
Decision Tree	0.599	0.599
Random Forest	0.656	0.714
Bagging	0.606	0.654
AdaBoost	0.641	0.694
SVM	0.607	0.638
KNN	0.547	0.668



Performance under the full variables models

Algorithms	Accuracy(1 - True Error Rate)	AUC
Logistic Regression	0.656	0.703
Decision Tree	0.622	0.641
Random Forest	0.668	0.729
Bagging	0.624	0.658
AdaBoost	0.660	0.713



Compare with reference

Algorithms	Accuracy(1 - True Error Rate)	AUC
Logistic Regression	0.636	0.680
Decision Tree	0.599	0.599
Random Forest	0.656	0.714
Bagging	0.606	0.654
AdaBoost	0.641	0.694
SVM	0.607	0.638
KNN	0.547	0.668

TABLE IV. PERFORMANCE OF DIFFERENT ALGORITHMS

Algorithms	Accuracy	Recall
Linear Regression	0.66	0.67
Logistic Regression	0.66	0.70
SVM ($d = 9$ Poly Kernel)	0.55	0.45
Random Forest (500 Trees)	**0.69**	**0.71**
k-Nearest Neighbors ($k = 5$)	0.56	0.47
SVR (Linear Kernel)	0.52	0.59
REPTree	0.67	0.62
Kernel Partial Least Square	0.58	0.60
Kernel Perceptron (Max loop 100)	0.45	0.99
C4.5 Algorithm	0.58	0.59

PART 05/Conclusion

Conclusion

Performance by Random Forest is the best among all the model we use.
We want to improve the popularity of articles according to the variable importance by Random Forest model.

- ✓ Number of words in article
- ✓ Number of images
- ✓ polarity

Thank you for listening