Predicting Popularity of Online News Reports

Nirmit Gupta (19) , Meet Chaudhary (44) , Bhavya Dubey (49) , Harman Singh (50)

Abstract — **With the help of Internet, the web news are often instantly spread around the world. Most of the people now have the habit of reading and sharing news online, for example , using social media like Twitter and Facebook. Typically, the news popularity are often indicated by the amount of reads, likes or shares. For the web news stake holders, it's very valuable if the recognition of the news articles are often accurately predicted before the publication. Thus, it's interesting and meaningful to use the machine learning techniques to predict the recognition of online news articles.In our project, the dataset including 39,643 news articles from website Mashable, we attempt to find the simplest classification learning algorithm to accurately predict if a news story will become popular or not before publication. Our research reflects that predicting news popularity with the help of tweets in social media platforms based on their shares and hashtags used during weekdays and weekends being done using different classification algorithms i.e linear regression, random forest, adaboost gives different result and the most accurate result obtained by our research is from Random forest classification algorithm giving an accuracy rate of 0.6743 which is highest amongst all the other classification algorithms we've used.**

Keywords— Random forest, Adaboost, logistic regression, Mashable, News popularity

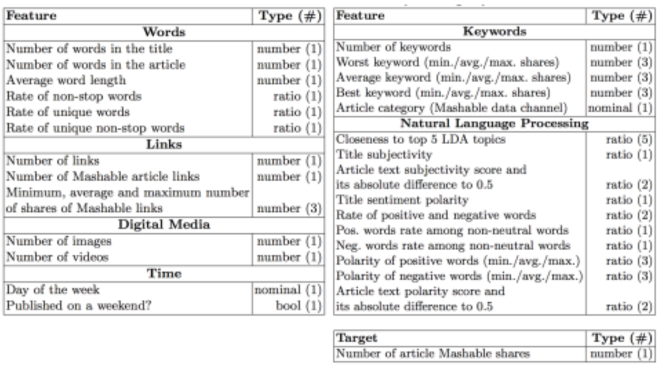
I. INTRODUCTION

Wo Working with machine learning algorithms within the large dataset is extremely common and particularly with the expansion of online news, it became very useful. Random Forest, linear regression and Adaboost are the common machine learning algorithms used for classification. during this research, we aimed to seek out the simplest model and set of features to predict the recognition of online news, using machine-learning techniques and implement various machine learning algorithms on the chosen features.The data source was Mashable, a well known online news website. Precision, AUC (Area Under the Curve) and F-measure were used to evaluate the results and their results were compared to seek out the foremost accurate amongst all. Random Forest seems to be the simplest model for prediction, and may achieve an accuracy of around 67%. Our work can help online news companies to predict news popularity before publication.

Various works have been done for prediction of online news popularity. In [1], the recognition of online articles is analyzed supported the user's comments. [2] defines the recognition in terms of a contest where the favored articles are those which were the foremost appealing thereon particular day. Ranking Support Vector Machine (SVM) is employed to classify the appealing/non appealing of online news story . In [3], the amount of retweets is predicted using both the features of the retweet content (length, words, number of hashtag, etc.) and therefore the features of author (number of followers, friends, etc.). [4] collects a dataset with almost 40,000 articles from the Mashable website, compares five different methods on classifying popular/unpopular news articles and concludes that the Random Forest (RF) are able to do the simplest performance.

I.I *Data Exploration*

The dataset is consisted of 39,643 news articles from an online news website called Mashable collected over 2 years from Jan. 2013 to Jan. 2015. It is downloaded from UCI Machine Learning Repository as https://archive.ics.uci.edu/ml/datasets/Online+News+Popularity# and this dataset is generously denoted by the author of [4]. For each instance of the dataset, it has 61 attributes which includes 1 target attribute (number of shares), 2 non-predictive features (URL of the article and Days between the article publication and the dataset acquisition) and 58 predictive features as shown in Fig. 1. The dataset has already been initially preprocessed. For examples, the categorical features like the published day of the week and article category have been transformed by one-hot encoding scheme, and the skewed feature like number of words in the article has been log transformed.



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)

Fig. 1. List of predictive attributes of dataset.

I.II *Data Visualization*

By observing various features, We think there are several relevant features like day of the week and article category. In Fig. 2, the count of popular/unpopular news over different days of the week is plotted. We can clearly find that the articles published over the weekends have larger potential to be popular. It makes sense because it is very likely that people will spend more time online browsing the news over the weekends. In Fig. 3, the count of popular/unpopular news over different article category is plotted. We can observe that in category of technology (”data channel is tech”) and social media (”data channel is socmed(social media”), the proportion of popular news is much larger the unpopular ones, and in category of world (”data channel is world”) and entertainment (”data channel is entertainment”), the proportion of unpopular news is larger than popular ones. This might reflect that the readers of Mashable prefer the channel of technology and social media much over the channel of world and entertainment.

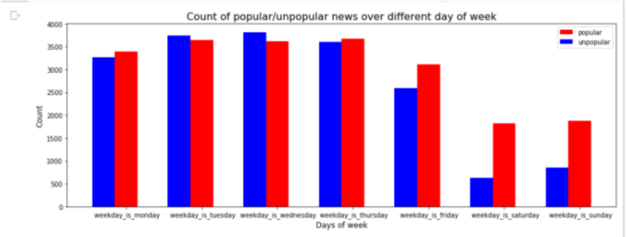


Fig. 2. Count of popular/unpopular news over different days of a week

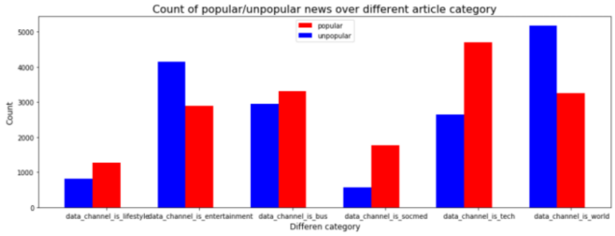
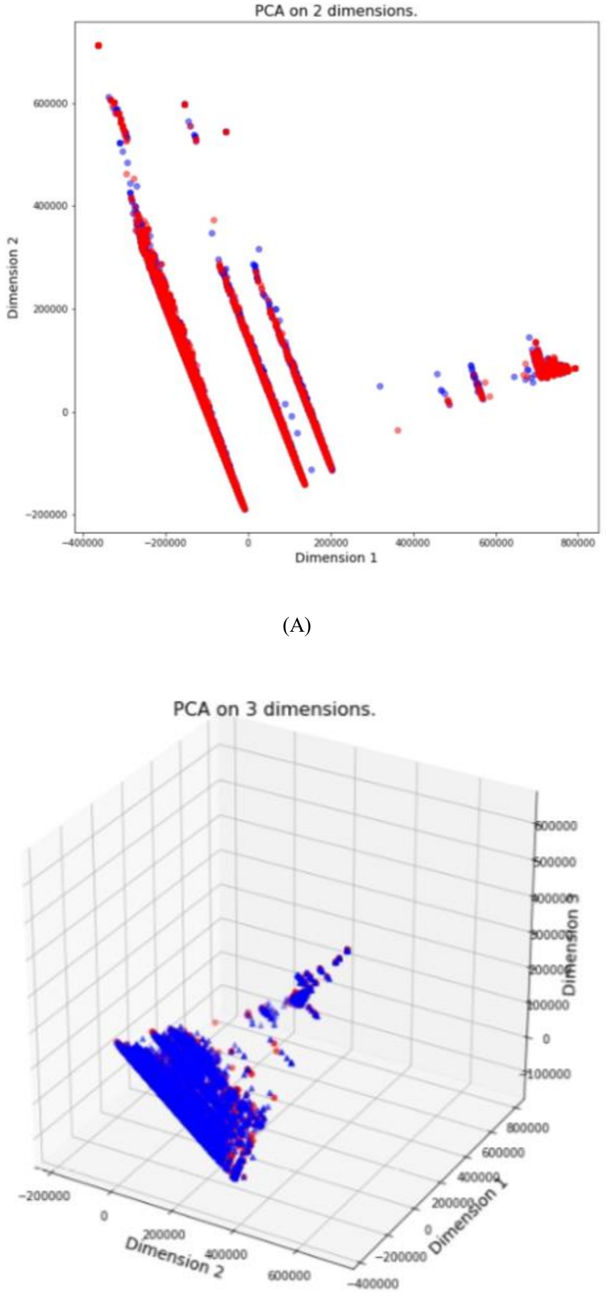


Fig. 3. Count of popular/unpopular news over different article category

Next, we do the principle component analysis (PCA) to visualize the data. As shown in Fig. 4, we project the data point onto first 2 and 3 principle components, respectively. It is clear that the dataset is not linearly separable in PCA space.



(A)

(B)

Fig. 4. (a) Data projected on first 2 principle components (b) Data projected on first 3 principle components. Red: popular; Blue: unpopular.

II. METHODOLGY

II.I Data Pre-Processing

As mentioned in Section 2.1, some data preprocessing works have been done by the data’s donator. The categorical features like the published day of the week and article category have been transformed by one-hot encoding scheme, and the skewed feature like number of words in the article has been log-transformed. Based on this, I further preprocess the dataset by normalizing the numerical feature to the interval [0, 1] such that each feature is treated equally when applying supervised learning. I also select the median of target attribute as the threshold to convert the continuous target attribute to boolean label.

Since there are 58 features in the dataset, it is reasonable to conduct a feature selection to reduce the data noise and increase the algorithm running speed. One effective way is using recursive feature elimination with cross validation (RFECV) to automatically select the most significant features for certain classifier.

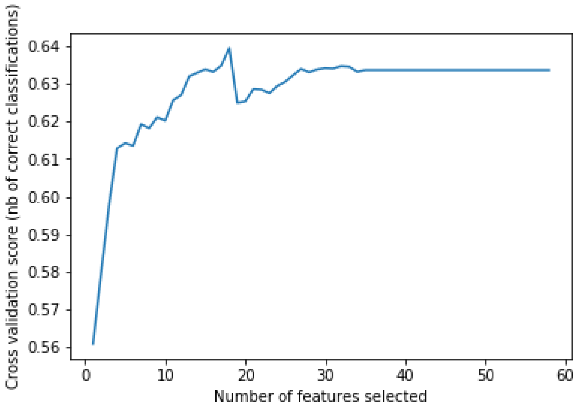


Fig. 5. The cross validation score versus the number of feature selected using RFECV for logistic regression estimator.

Firstly, we run RFECV with a logistic regression estimator. The cross validation score versus the number of feature selected is shown in Fig. 6. From the figure, we can find there is drop of score when number of features is 29. Thus RFECV algorithm select 29 most relevant features from 58 original features. The selected 29 features are listed in Fig. 7. Interestingly, the day of week and article category features are included in these 29 features.

Next, we run RFECV with a RF estimator. The cross validation score versus the number of feature selected is shown in Fig. 8. We find that RFECV selects 56 features for RF, which is almost the full features in dataset. The only two features RFECV excludes for RF are [’ n non stop words’, ’ data channel is lifestyle’].

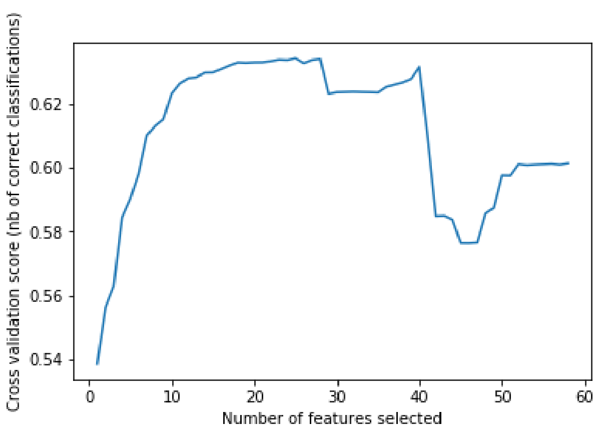


Fig. 6. The cross validation score versus the number of feature selected using RFECV with RF estimator.

Lastly, we run RFECV with Adaboost estimator. The cross validation score versus the number of feature selected is shown in Fig. 9. From the figure, we can find there is a peak when number of features is 18. Thus RFECV algorithm select 18 most relevant features from 58 original features.

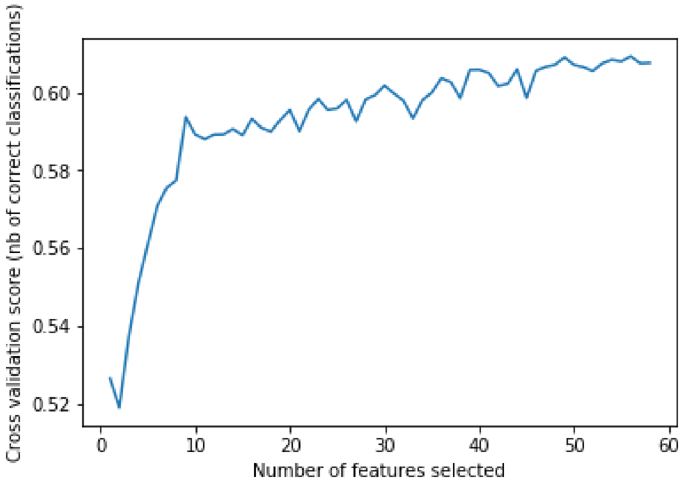


Figure 7: The cross validation score versus the number of feature selected using RFECV with Adaboost estimator.

II.II *Implementation*

Before algorithm implementation, for each algorithm, I also randomly split dataset with its own selected features into training set (90%) and testing set (10%). The logistic regression, RF and Adaboost are implemented by the sklearn function LogisticRegression(), RandomForestClassifier() and AdaBoostClassifier(), respectively. At this stage, I will first use the default setting for model hyperparameters. The default hyperparameters are logistic regression - {”C”: 1.0}; RF-{”n estimators”: 10}; Adaboost-{”n estimators”: 50, ”learning rate”: 1.0}. I will try to refine the parameters in next part. With the selected feature for each algorithm, I run the three classification algorithms and their performance is shown in Fig.

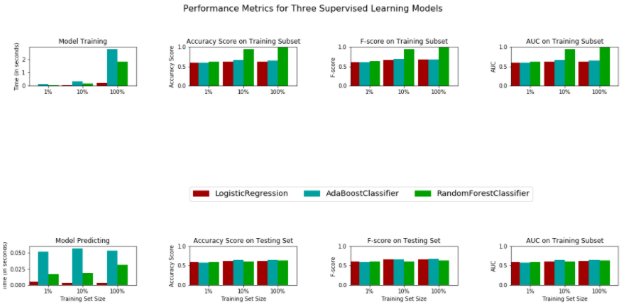


Fig. 8. Performance of three classifiers under default parameter setting.

Classifier	Accuracy	F1-Score	AUC
Logistic Regression	0.633	0.6767	0.6281
Random Forest	0.6769	0.7073	0.6734
Adaboost	0.6567	0.6873	0.6535

Fig.Fig.9. Metrics score of three classifiers under refined hyperparameter

The three metrics (accuracy, F1-score and AUC) are summarized in Fig 12. Under default parameter setting,

Adaboost performs best in all three metrics, RF performs better than logistic regression in AUC while logistic regression performs better than RF in accuracy and F1-score. As for the training and testing speed, logistic regression is much faster than the other two, and RF runs faster than Adaboost.

III. Experimental Results

After initial implementation and further refinement for the three classifiers, we find the best performance is obtained by the RF classifier with 500 trees in the forest. The best obtained metrics of RF has the accuracy of 0.6769, F1-score 0.7073 and AUC 0.6734. The final scores are not exceptional, which is sort of within the expectation, because the dataset is not linear separable as shown in the PCA in Section 2.1. But it still achieves a reasonable performance in news popularity prediction compared with a random guess. To test the robustness of the model, we change the split ratio of training/testing set from 0.1 to 0.15, and then run the three classifier with the same refined hyperparameters. Now the metrics are shown in Fig. 10. Compared with Fig. 9, the performance of the model is still similar and the best performance is still given by RF.

.

Classifier	Accuracy	F1-Score	AUC
Logistic Regression	0.633	0.6767	0.6281
Random Forest	0.6769	0.7073	0.6734
Adaboost	0.6567	0.6873	0.6535

Fig. 10. Test the model with training/testing set ratio 0.15

IV. Conclusion

We come to conclusion after comparing the results obtained from all the three classifiers used that Random forest algorithm proves to be the most accurate amongst all giving us an accuracy rate of 67%. The training and testing time of RF classifier is greatly increased since 500 trees are used in the forest, but it helps RF achieve the best performance in terms of accuracy, F1-score and AUC.

References

The template will number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use “Ref. [3]” or “reference [3]” except at the beginning of a sentence: “Reference [3] was the first ...”

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors’ names; do not use “et al.”. Papers that have not been published, even if they have been submitted for publication, should be cited as “unpublished” [4]. Papers that have been accepted for publication should be cited as “in press” [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

[1] A. Tatar, J. Leguay, P. Antoniadis, A. Limbourg, M. D. de Amorim, and S. Fdida, “Predicting the popularity of online articles based on user comments,” in Proceedings of the International Conference on Web Intelligence, Mining and Semantics. ACM, 2011, p. 67.

[2] E. Hensinger, I. Flaounas, and N. Cristianini, “Modelling and predicting news popularity,” Pattern Analysis and Applications, vol. 16, no. 4, pp. 623–635, 2013.

[3] S. Petrovic, M. Osborne, and V. Lavrenko, “Rt to win! predicting message propagation in twitter.” ICWSM, vol. 11, pp. 586–589,

2011. [4] K. Fernandes, P. Vinagre, and P. Cortez, “A proactive intelligent decision support system for predicting the popularity of online news,” in Portuguese Conference on Artificial Intelligence. Springer, 2015, pp. 535–546.

IEEE xxxx