REPORT ON
MACHINE LEARNING PROJECT
ON
ONLINE NEWS POPULARITY PREDICTION

Student Name: RHYTHM KULSHRESTHA

Enrolment Number: 09819011921

Email ID: rkul0103@gmail.com

Contact Number: 9811887743

Code:

https://colab.research.google.com/drive/1YJ_Afr7krBr7tkoLpuo-HYT4vmMakI_E?usp=sharing

Data Set:

Online News Popularity - UCI Machine Learning Repository

Google Website Link:

https://sites.google.com/view/onlinenewspopularity/home

Youtube video link:

https://youtu.be/Wq7dWVDrqpc

INTRODUCTION

The rise of the internet has enabled rapid global dissemination of online news, leading to the widespread habit of reading and sharing news articles through platforms like Twitter and Facebook. The popularity of online news is often measured by metrics such as the number of reads, likes, or shares. Content providers and advertisers who rely on online news are keen to accurately predict the popularity of articles before publication. Consequently, using machine learning techniques to forecast the popularity of online news articles has become an interesting and meaningful area of research.

PROBLEM STATEMENT

This project focuses on utilizing machine learning techniques to address a binary classification problem related to online news articles. The main objective is to predict whether an article will achieve popularity before its publication, based on the number of shares it receives. To achieve this, the popularity of an article is measured by comparing the number of shares to a predetermined threshold. If the number of shares exceeds the threshold, the article is considered popular; otherwise, it is labeled as unpopular. The project aims to leverage a set of features associated with the articles and determine the most effective machine learning model for accurately classifying the target label (popular or unpopular) of articles that have not yet been published.
In order to accomplish this task, 6 classification learning algorithms will be implemented and compared: Logistic Regression, Random Forest (RF),Decision Tree, Bagging, KNN and Adaboost. These algorithms will be evaluated based on various metrics to assess their performance and accuracy in predicting the popularity of articles. By comparing the results of these models, the best-performing algorithm will be selected to serve as the primary predictor of article popularity.

The significance of this project lies in its potential to assist stakeholders in the online news domain, such as content providers and advertisers, by offering valuable insights into the future popularity of articles. By accurately predicting the popularity of articles prior to publication, these stakeholders can make informed decisions regarding content selection, marketing strategies, and resource allocation.

Machine learning techniques provide a powerful toolset for analyzing and understanding the factors that contribute to the popularity of online news articles. By leveraging various features and attributes associated with news articles, such as textual content, multimedia elements, publication timing, and more, machine learning models can be trained to identify patterns and make predictions about an article's potential popularity. This only assists in strategic decision-making but also helps content providers stay ahead of the competition by offering content that resonates with their target audience.

DATASET DESCIPTION

The dataset is consists 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 .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. 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 have been log transformed.There are no missing values in the dataset.

METRICS

As a classification task, we will adopt the following three evaluation metrics: accuracy,F1-score and AUC. For all three metrics, the higher value of the metric means the better performance of model.

a) Accuracy: Accuracy is direct indication of the proportion of correct classification. It considers both true positives and true negatives with equal weight and it can be computed as accuracy = true positives + true negatives dataset size . Although the measure of accuracy might be naive when the data class distribution is highly skewed, but it is still an intuitive indication of model’s performance.

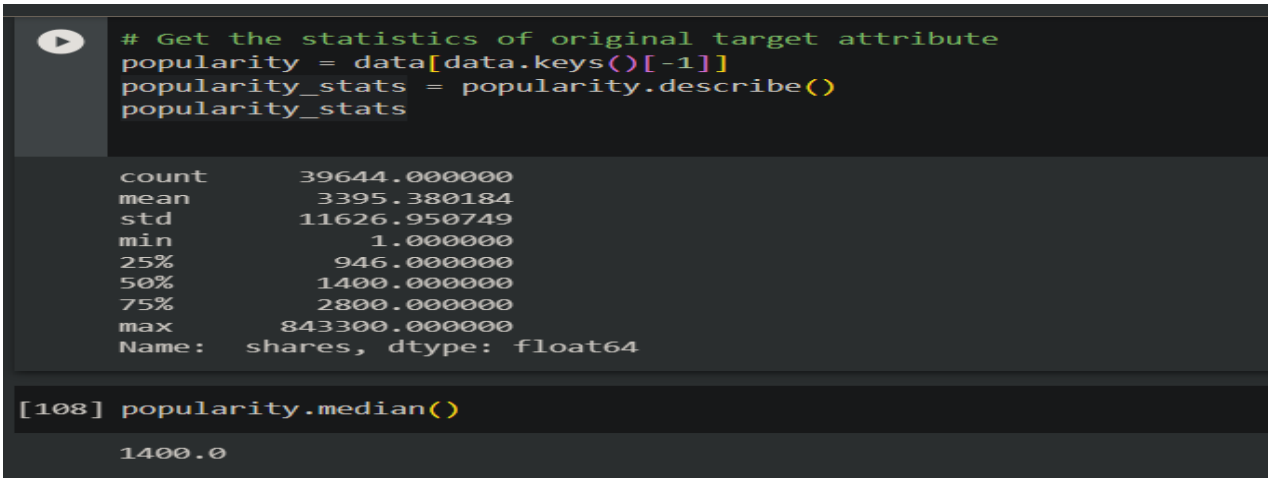
b)F1-score: F1-score is an unweighted measure for accuracy by taking harmonic mean of precision and recall, which can be computed as F1 = 2 · precision · recall precision + recall (2) It is a robust measurement since it is independent of data class distribution.

c) AUC: The AUC is the area under the ROC (Receiver Operating Characteristics) curve, which is a plot of the True Positive Rate versus the False

Positive Rate. AUC value is a good measure of classifier's discrimination power and it is a more robust measure for model performance.

EXPLORATORY DATA ANALYSIS

Firstly, we need to determine the appropriate threshold for number of shares to discriminate the news to be popular or unpopular. So we show the statistics of the target attribute ”shares” and we find the median of target attribute is 1,400, thus it is reasonable if we take 1,400 as a threshold. Then we can use this threshold to convert the continuous number target attribute into a boolean label.



```
# Get the statistics of original target attribute
popularity = data[data.keys()[-1]]
popularity_stats = popularity.describe()
popularity_stats

count      39644.000000
mean        3395.380184
std        11626.950749
min            1.000000
25%          946.000000
50%         1400.000000
75%         2800.000000
max       843300.000000
Name: shares, dtype: float64

[108] popularity.median()

1400.0
```

DROP NON-PREDICTIVE COLUMNS

After this, features_raw = data.drop(['url',data.keys()[1],data.keys()[-1]], axis=1) is used to create a new dataset called features_raw. This dataset is derived from the original dataset called data, but with certain columns removed. Specifically, the columns with the label 'url', the second column, ‘timedelta’ are removed because they are non-predictive, and the target column ‘shares’ is dropped.

FEATURE SCALING

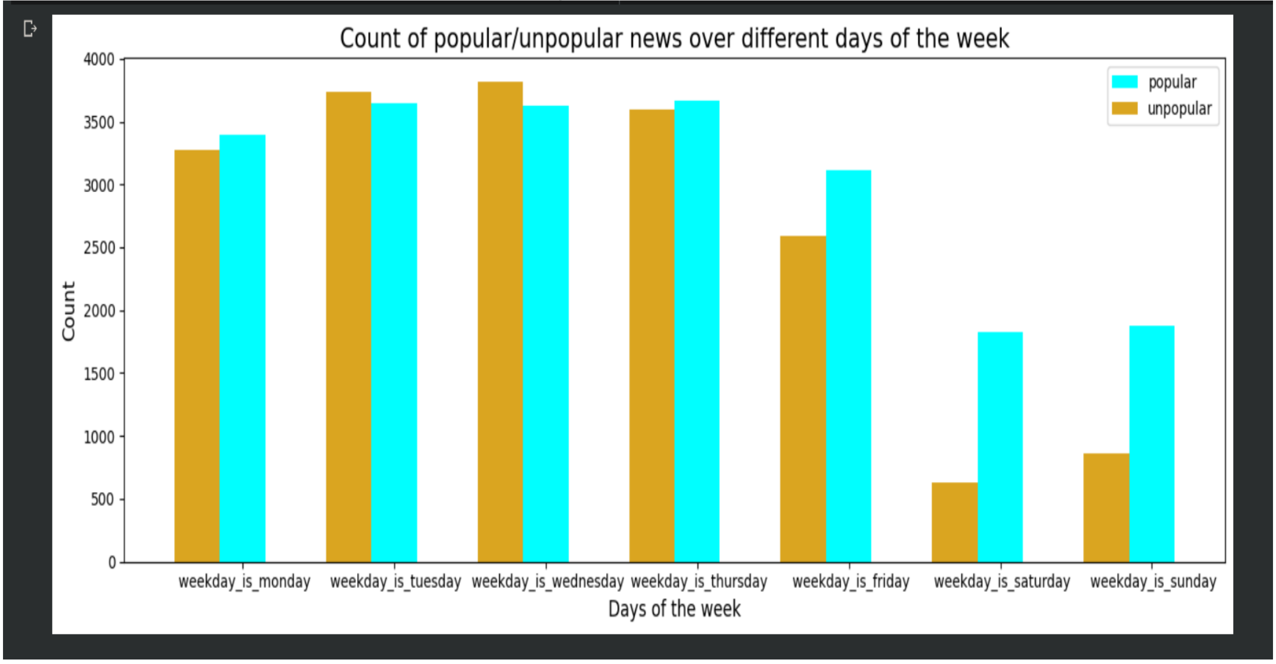
Thenafter,we normalize the numerical features in the dataset. Normalization is a process that transforms numerical data to a standard scale, making it easier to compare and analyze the data.

In this case, the StandardScaler from the sklearn.preprocessing module is utilized for normalization. The StandardScaler object is created, which will be used to scale the numerical features.The list numerical contains the names of the numerical features that need to be normalized.The scaler.fit_transform(data[numerical]) part of the code applies the scaling transformation to the specified numerical features in the original data dataset. It calculates the mean and standard deviation of each feature and scales the values accordingly.

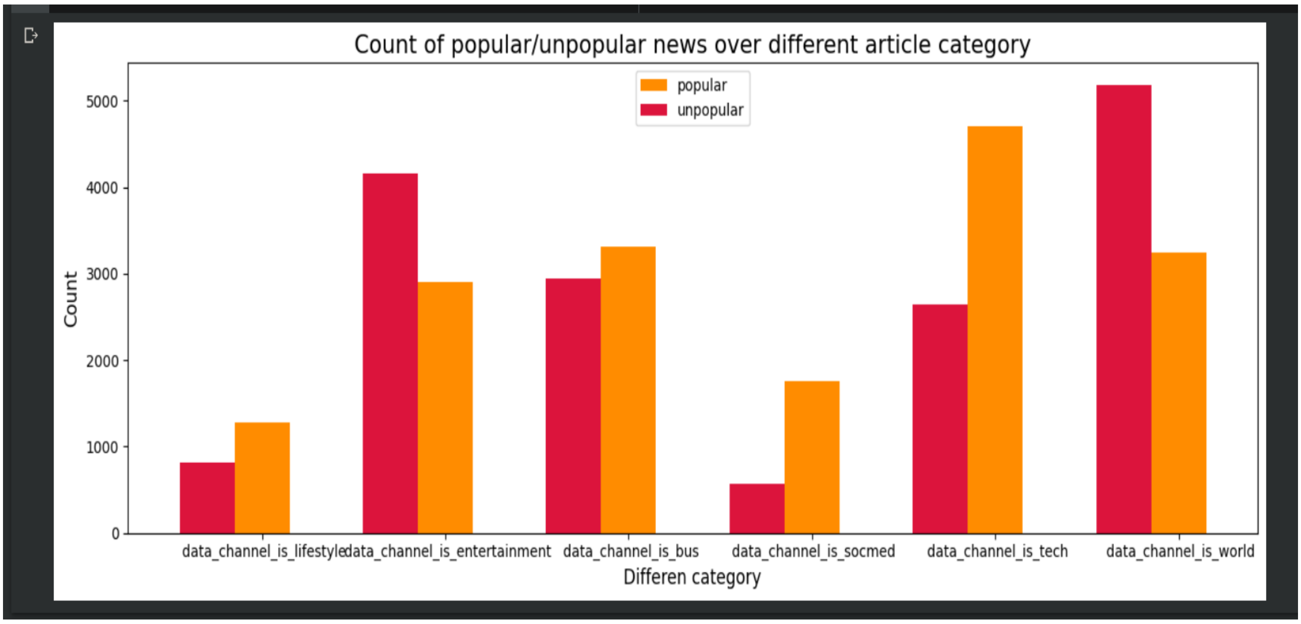
DATA VISUALISATIONS

By observing and analysing the plot of different featues with respect to number of shares ,there seems to be a relation between them:

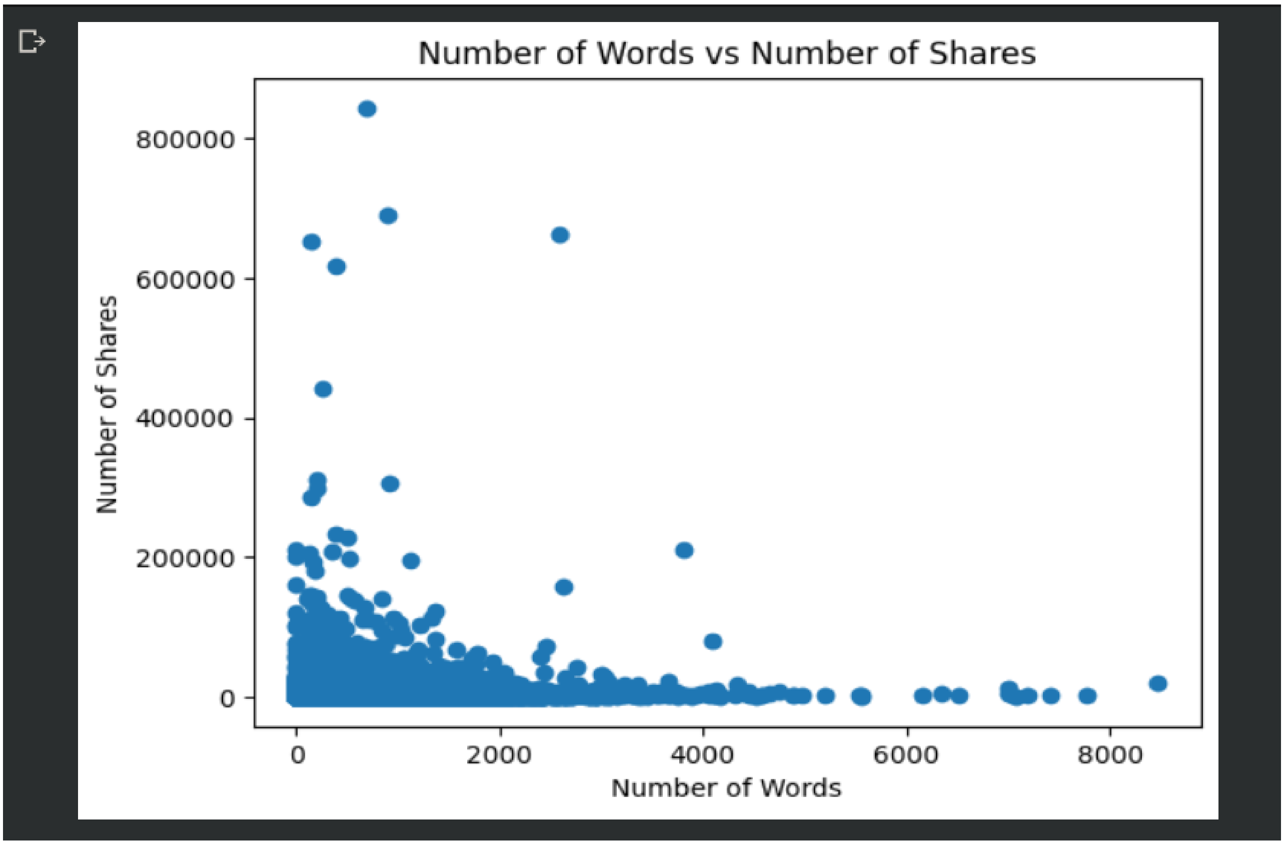
1. The count of popular/unpopular news over different day of the week is plotted. We can clearly find that the articles published over the weekends has larger potential to be popular. It makes sense because it is very likely that people will spend more time online browsing the news over the weekends.



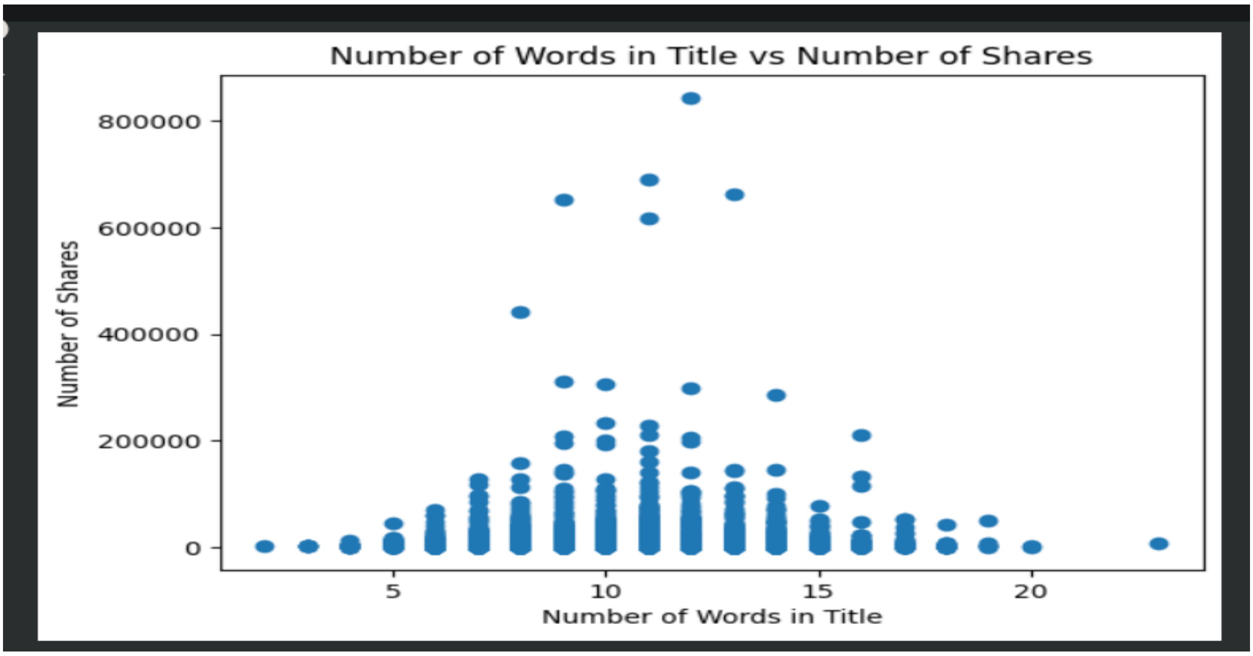
2.The count of popular/unpopular news over different article category is plotted. We can observe that in category of technology (”data channel is tech”) and social media (”data channel is socmed”), the proportion of popular news is much larger the unpopular ones, and in category of world (”data channel is world”) and entertainment (”data channel is entertainment”), the proporttion of unpopular news is larger the 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.



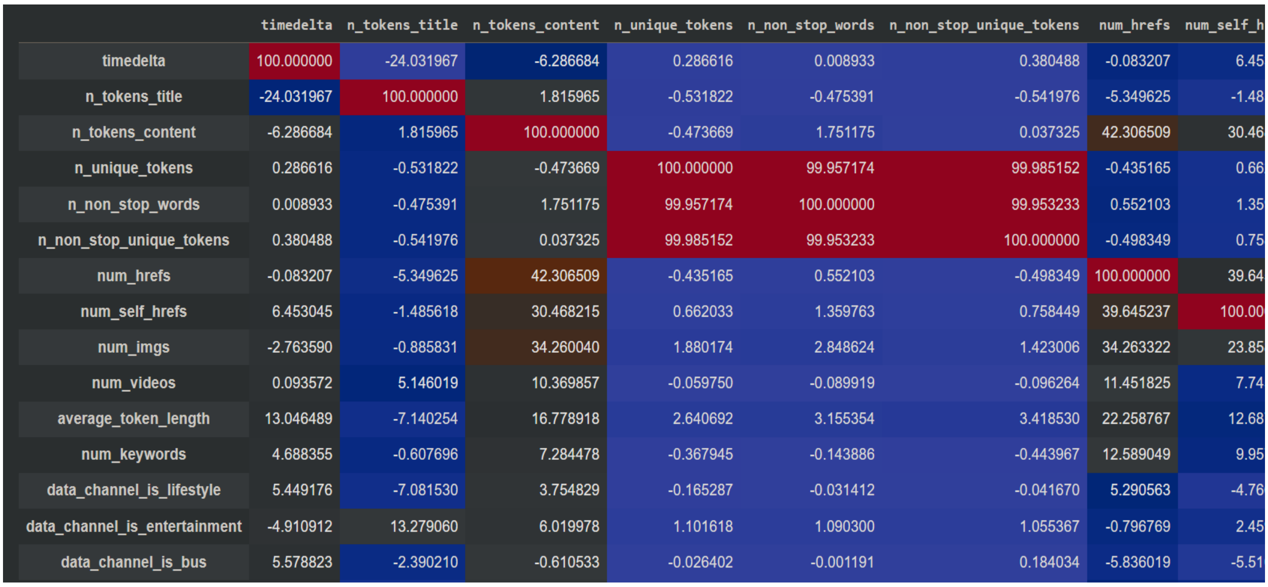
3)A scatter plot of number of words vs Number of shares suggests that number of shares are more when the the number of words are less.



4)A scatter plot of Number of Words in Title Vs Number of shares suggests that shares are somewhat normally distributed with mean at 12.3



5)We have calculated the correlation between the variables in the dataset and multiplied the values by 100 to make them more easily interpretable. Correlation measures the strength and direction of the relationship between two variables. Higher positive values indicate a strong positive relationship, while higher negative values indicate a strong negative relationship. The style.background_gradient('coolwarm') part of the code adds a color gradient to visually represent the correlation values, with cooler colors (like blue) representing negative correlations and warmer colors (like red) representing positive correlations. This helps us visually identify patterns and relationships between variables in the dataset.



	timedelta	n_tokens_title	n_tokens_content	n_unique_tokens	n_non_stop_words	n_non_stop_unique_tokens	num_hrefs	num_self_h
timedelta	100.000000	-24.031967	-6.286684	0.286616	0.008933	0.380488	-0.083207	6.45
n_tokens_title	-24.031967	100.000000	1.815965	-0.531822	-0.475391	-0.541976	-5.349625	-1.48
n_tokens_content	-6.286684	1.815965	100.000000	-0.473669	1.751175	0.037325	42.306509	30.46
n_unique_tokens	0.286616	-0.531822	-0.473669	100.000000	99.957174	99.985152	-0.435165	0.66
n_non_stop_words	0.008933	-0.475391	1.751175	99.957174	100.000000	99.953233	0.552103	1.35
n_non_stop_unique_tokens	0.380488	-0.541976	0.037325	99.985152	99.953233	100.000000	-0.498349	0.75
num_hrefs	-0.083207	-5.349625	42.306509	-0.435165	0.552103	-0.498349	100.000000	39.64
num_self_hrefs	6.453045	-1.485618	30.468215	0.662033	1.359763	0.758449	39.645237	100.00
num_imgs	-2.763590	-0.885831	34.260040	1.880174	2.848624	1.423006	34.263322	23.85
num_videos	0.093572	5.146019	10.369857	-0.059750	-0.089919	-0.096264	11.451825	7.74
average_token_length	13.046489	-7.140254	16.778918	2.640692	3.155354	3.418530	22.258767	12.68
num_keywords	4.688355	-0.607696	7.284478	-0.367945	-0.143886	-0.443967	12.589049	9.95
data_channel_is_lifestyle	5.449176	-7.081530	3.754829	-0.165287	-0.031412	-0.041670	5.290563	-4.76
data_channel_is_entertainment	-4.910912	13.279060	6.019978	1.101618	1.090300	1.055367	-0.796769	2.45
data_channel_is_bus	5.578823	-2.390210	-0.610533	-0.026402	-0.001191	0.184034	-5.836019	-5.51

FEATURE SELECTION

We perform feature selection using the SelectKBest method in scikit-learn. It selects the top 30 features from the dataset based on their F-statistic scores, which measure the linear dependency between each feature and the target variable. The transformed dataset contains only the selected features. The selected feature indices and names are retrieved, providing insights into the most important features for the classification task. This feature selection step helps reduce dimensionality and focuses on the most relevant features, potentially improving the model's performance and interpretability. The printed output displays the names of the selected features, allowing further analysis and examination of their significance in predicting the popularity of online news articles.

ALGORITHMS AND TECHNIQUES

We formulate this problem as a binary classification problem . In this project, six classification learning algorithms including Logistic Regression, RF,DecisionTree,KNN,bagging and Adaboost will be implemented and compared based on the evaluation metric such as accuracy,precision, F1-score and Area Under ROC Curve (AUC).

Before algorithm implementation, for each algorithm, we randomly split dataset with its own selected features into training set (90%) and testing set (10%)

We train and evaluate multiple classifiers on the given data. Various classification models such as Random Forest, Logistic Regression, K-Nearest Neighbors, Decision Tree, AdaBoost, and Bagging are stored in a dictionary called classifier.

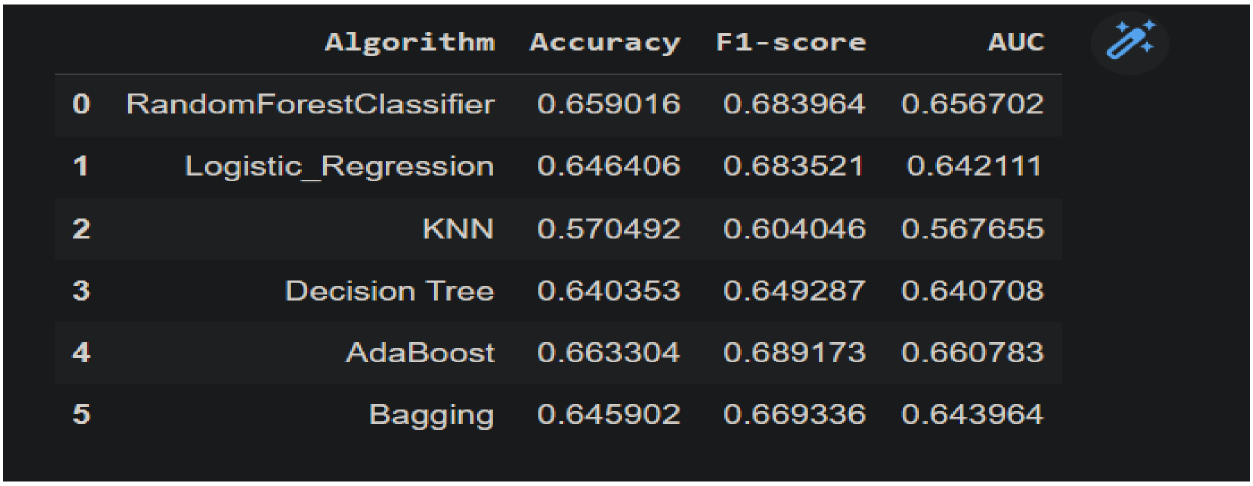
The train_classifier function is defined to train and evaluate a given classifier on the training and test sets. It fits the classifier to the training data, predicts the labels for the test data, and calculates evaluation metrics such as accuracy, precision, F1-score, and AUC-ROC score.

The loop iterates over the classifiers in the classifier dictionary and calls the train_classifier function for each classifier. It prints the evaluation metrics for each model, including accuracy, precision, F1-score, and AUC-ROC score.

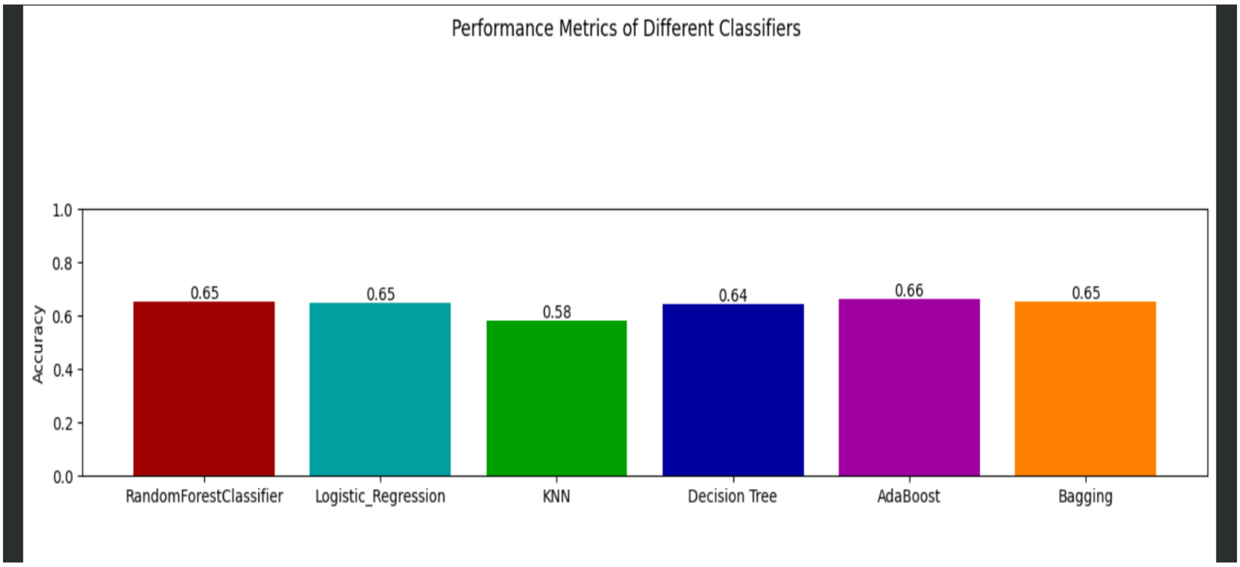
The accuracy, precision, F1-score, and AUC-ROC score for each model are stored in separate lists (accuracy_score1, precision_score2, f1_score3, auc_score4). Finally, these metrics are used to create a Pandas DataFrame called performance_dp, which summarizes the performance of each classifier, including the algorithm name and the corresponding evaluation scores.

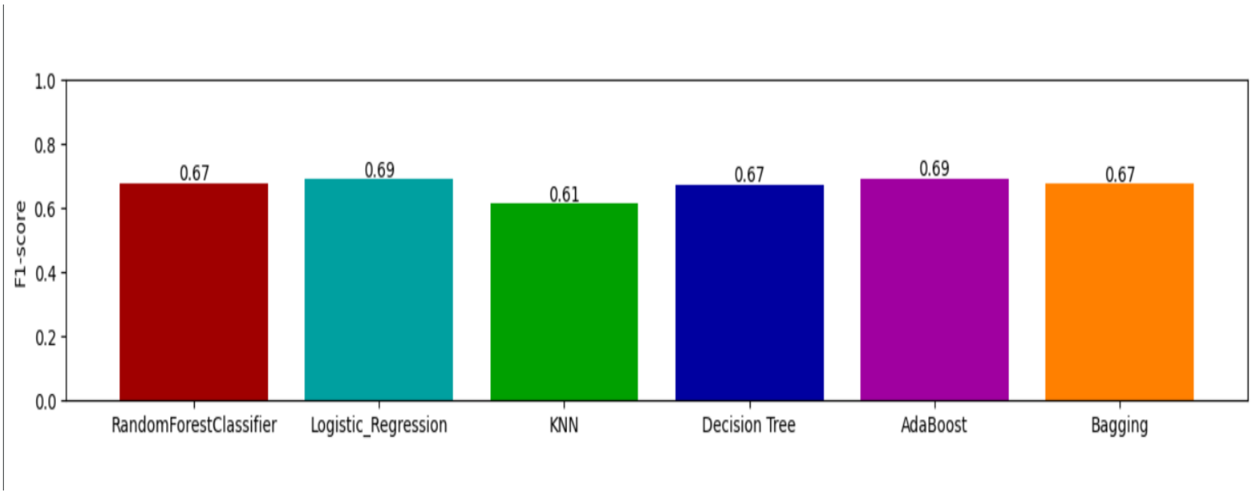
RESULTS

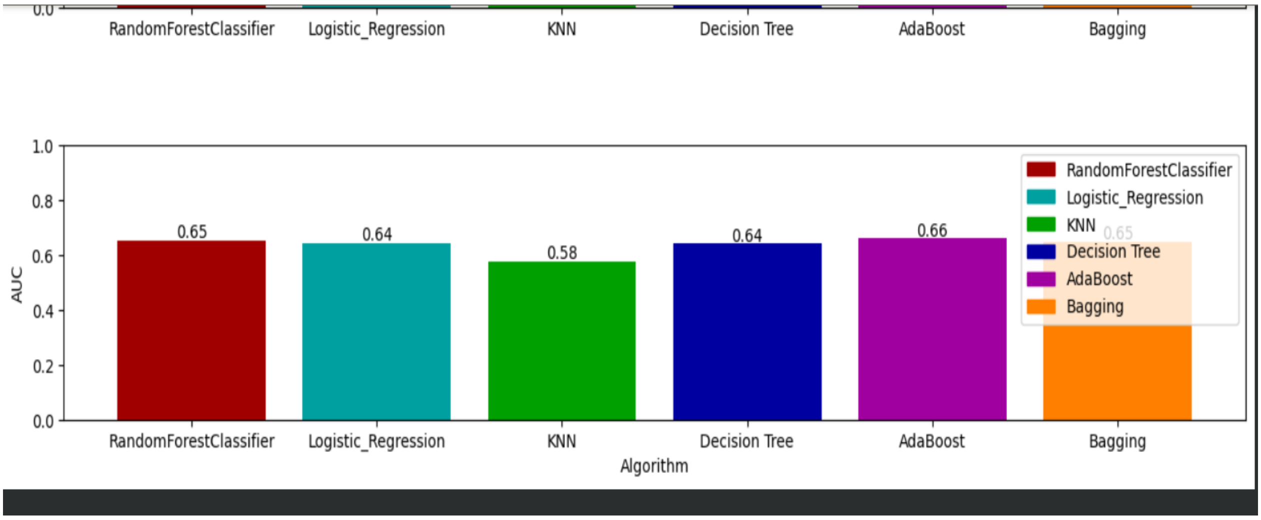
After implementation and analysis for the the classifiers, we find the best performance is obtained by the Adaboost classifier. The best obtained metrics of adaboost are accuracy 0.66330,F1-score 0.689173,AUC score 0.660783.



	Algorithm	Accuracy	F1-score	AUC
0	RandomForestClassifier	0.659016	0.683964	0.656702
1	Logistic_Regression	0.646406	0.683521	0.642111
2	KNN	0.570492	0.604046	0.567655
3	Decision Tree	0.640353	0.649287	0.640708
4	AdaBoost	0.663304	0.689173	0.660783
5	Bagging	0.645902	0.669336	0.643964







CONCLUSION

In conclusion, the online news popularity project aimed to predict the popularity of news articles prior to publication using machine learning techniques. By analyzing various features of the articles and their relationship with the number of shares on social media platforms, valuable insights can be gained for content providers and advertisers.

Throughout the project, several important steps were performed. The dataset was explored and preprocessed, including handling missing values, normalizing numerical features, and selecting relevant features. Feature selection techniques, such as SelectKBest, were applied to identify the most informative features for prediction.

Different classification algorithms, including Logistic Regression, Random Forest, AdaBoost, and others, were trained and evaluated using evaluation metrics such as accuracy, F1-score, and AUC-ROC score. These metrics provided a comprehensive assessment of the models' performance in predicting the popularity of news articles.

Among the classifiers, the AdaBoost algorithm demonstrated the best performance, indicating its effectiveness in accurately classifying popular and unpopular news articles. The AdaBoost algorithm combines weak classifiers to create a strong ensemble model, enhancing its predictive capabilities.

The project's findings have practical implications for content providers and advertisers, as the ability to predict the popularity of online news articles can inform decision-making processes regarding content promotion, advertising strategies, and resource allocation. By leveraging machine learning techniques, stakeholders can optimize their efforts to maximize the reach and impact of news articles.

It is worth noting that the success of the project relies on the quality and representativeness of the dataset, the selection of relevant features, and the chosen machine learning algorithms. Further research and experimentation may be necessary to refine and improve the predictive models for specific contexts and domains

REFERENCES

K. Fernandes, P. Vinagre and P. Cortez. A Proactive Intelligent Decision support System for Predicting the Popularity of Online News. Proceedings of the 17th EPIA 2015 - Portuguese Conference on Artificial Intelligence,September, Coimbra, Portugal.