Prediction of Online News Popularity using Supervised and Unsupervised Machine Learning Techniques

Abhishek Sushil
abhishek21441@iiitd.ac.in

Anurag Gupta
anurag21451@iiitd.ac.in

Ayush Srivastava
ayush21457@iiitd.ac.in

Manas Narang
manas21473@iiitd.ac.in

1. Motivation

With the rapid growth of digital media, understanding and predicting the popularity of online news articles is crucial for content creators, marketers, and media outlets. Accurately forecasting the popularity of news articles can help in optimizing content distribution, enhancing engagement, and tailoring marketing strategies.This data set encapsulates a diverse array of attributes pertaining to articles published by Mashable over a span of two years.

2. Introduction: Problem Statement

In the context of our dataset, which contains information about Mashable articles, our primary objective is to predict the number of shares an article will receive. However, this prediction task is challenged by the intrinsic unpredictability and substantial variance in the factors contributing to articles achieving exceptionally high levels of virality, typically defined as surpassing 100,000 shares.

In light of these challenges, we have opted to transform the conventional regression problem into a classification problem. This transformation entails categorizing the continuous share data into distinct classes, each representing a range of popularity based on the percentiles of shares received. As a result, our goal shifts from predicting precise share counts to determining the relative popularity category of an article, providing a more robust and actionable framework for our analysis.

By adopting this approach, we are better equipped to manage the unpredictability and wide-ranging factors that influence an article's virality. Our focus shifts from the exact number of shares, which can be highly variable and challenging to predict accurately, to a more interpretable and manageable system of popularity classes. This allows us to make informed assessments of an article's expected popularity within a reasonable range, thereby enhancing the utility of our predictive model.

3. Literature Survey

The paper by Talwar[1] used for binary classification along with their characteristics and techniques applied. Here's a breakdown of each model and its key points:

A. Linear Regression - Used for binary classification by converting the predicted number of shares to a classification result based on a threshold. - All 58 predictive features were used. - Assumes that features are mutually independent and that classes are linearly separable. - Not the most suitable model for a classification problem.

B. Logistic Regression - Feature selection using Recursive Feature Elimination (RFE) to retrieve the top 20 features. - Utilizes a non-linear activation function (sigmoid) to provide contrast to linear separability assumptions. - Regularization parameter with L2 regularization.

C. Support Vector Machine (SVM) - Feature selection using RFE to retrieve the top 20 features. - Trained with a radial basis function (RBF) kernel and a regularization parameter (C = 10). - Experimented with various kernels and chose RBF for best accuracy. - Provides a good generalization performance but has a longer computation time and does not return probabilistic confidence.

D. Random Forest - Utilized 100 decision trees and considered a maximum of 6 features at each decision node. - An ensemble classifier known for good generalization and non-overfitting. - Learns non-linear hypotheses and performs well.

E. Adaboost Classifier - Used decision stumps as base estimators. - Performed grid-search on the number of decision stumps and found the best performance with 100 decision stumps. - Combines bagging and boosting for learning complex non-linear hypotheses. - Achieved the best performance among the classifiers.

F. K-Nearest Neighbors (KNN) - Determines the class of each test sample based on the majority vote from its k nearest training samples. - Selected the appropriate k value for the best AUC score through a search. - Robust to fluctuations in feature values, using L2-Norm for similarity measurement. - Makes an assumption about the distance metric to use, which may vary based on the features.

The paper provided an excellent overview of the models, their characteristics, and the techniques applied, making it clear how each model was configured and its strengths and weaknesses in the context of binary classification.

4. Dataset

This dataset summarizes a heterogeneous set of features about articles published by Mashable in a period of two years. The goal is to predict the number of shares in social networks (popularity). - The dataset contains various features extracted through natural language process techniques. Including the unique and non unique tokens in the title, article, metadata.
- They also include the number of images, videos, hyperlinks to both Mashable and non-Mashable articles.
- Various categories of the maximum, minimum and mean shares of the best and worst keyword have been analysed as well.
- Global Subjectivity scores have been provided for each article that determine the performance of opinion pieces versus reporting facts. Several metrics for Global Sentiment Polarity have been provided as well, such as rate of positivity and negativity and the maximum, minimum and mean values for the same.
- The day of the week of article publishing has been provided and whether the article was published on a weekend or weekday.
- Finally Latent Dirichlet Allocation was performed that subdivided the data into 5 topics.

5. Methodology

5.1. Selecting the number of classes

The conversion of a continuous output class into a discrete form involved employing the K-means clustering technique within the methodology. The pivotal step was determining the optimal value for K by analyzing the within-cluster sum of squares (WCSS) graph to identify the point where the graph exhibited an elbow. This analysis led to the selection of K=4 as the optimum value for clustering. The adoption of this value segmented the continuous output class into four distinct and discrete categories. This methodological approach was pivotal in discretizing the initial continuous data, providing a structured framework for subsequent analysis and interpretation within the academic report.

5.2. Transforming output data to Classes

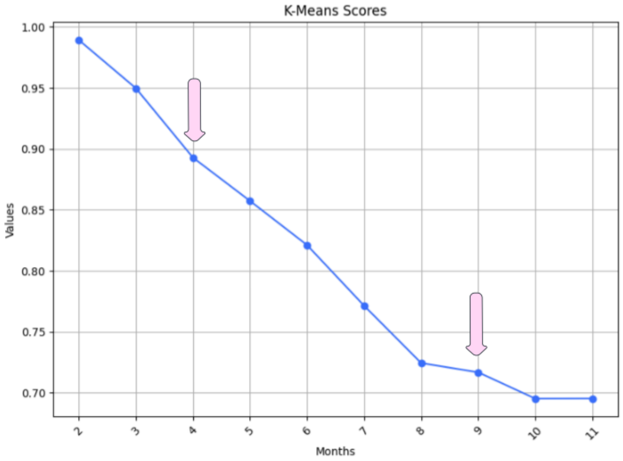

Figure 1. kmeans graph

Since the values in the shares column were continuous values, we needed to define some classes in order to transfrom this into a classification problem. We have used a percentile based approach to divide our data into 4 distinct classes, based on the number of times the articles were shared. The top 25% articles in one class, the next 25% in another and so on.

5.3. Removing outliers

By the histogram of the data, we can see that the dataset has some very low values and some very large values. Hence, we have removed 5% of samples with the lowest values of shares, and 5% of samples with the highest, to interpret the actual distribution of the sample values.

5.4. Feature Analysis and Selection

5.4.1 Removing non-predictive attributes

URL: URL of the article and **timedelta**: Days between the article publication and the dataset acquisition are non-predictive attributes and are just used for identification.

Hence, they have been removed from the dataset.

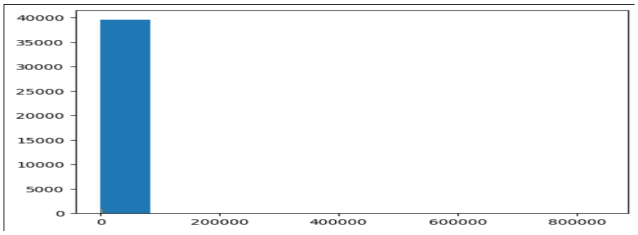

Figure 2. Org Hist

5.4.2 Dealing with Weekdays and Weekends

All the weekdays perform equally after removing the outliers. Hence, we remove the variables of the type **weekday_is_day**. The attribute **is_weekend** gives enough infor-

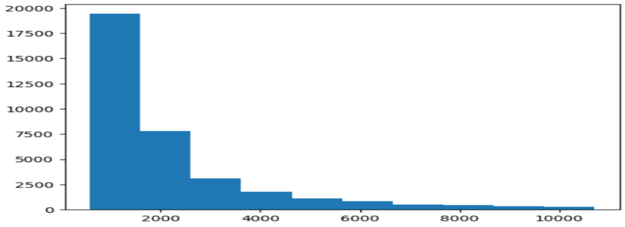

Figure 3. Without Outliers

mation to interpret the weightages of days in predicting the number of shares.

5.4.3 Subjects of Article

We plotted the number of shares of each category and observed their distribution and values at each quartile, judging sufficient and high virality at 75th and 95th percentiles, we found the following. Articles perform well in this order based on the subjects.

Life $\geq$ $Medicine \geq Tech \geq Entertainment \geq Business \geq World$

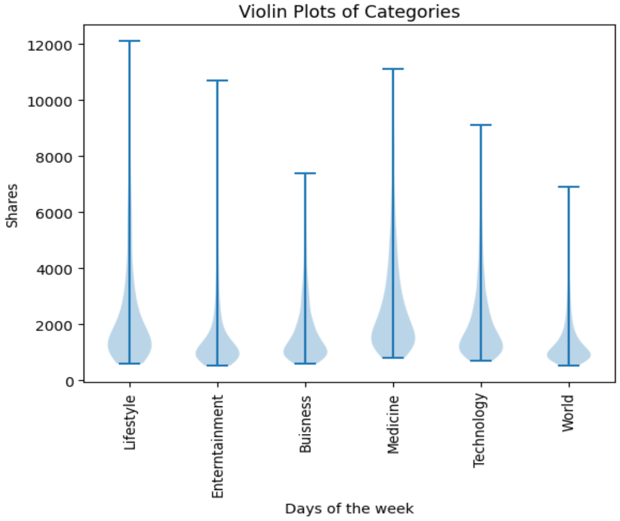

Figure 4. Violin Plot of Subjects

5.4.4 Removing the average attributes due to high correlation

We can see in multiple fields like keyword share rate, polarity rate, we can see that three attributes - max, min, as well as average are given. However, in these cases, average has high correlation (above 0.5) with the min-max attributes, which is not desirable. Hence, we have removed the average attributes and left the min-max attributes.

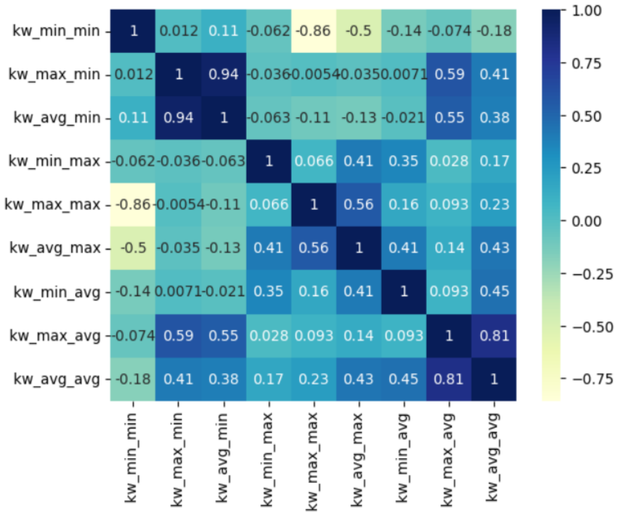

Figure 5. Covariance Map for Keywords

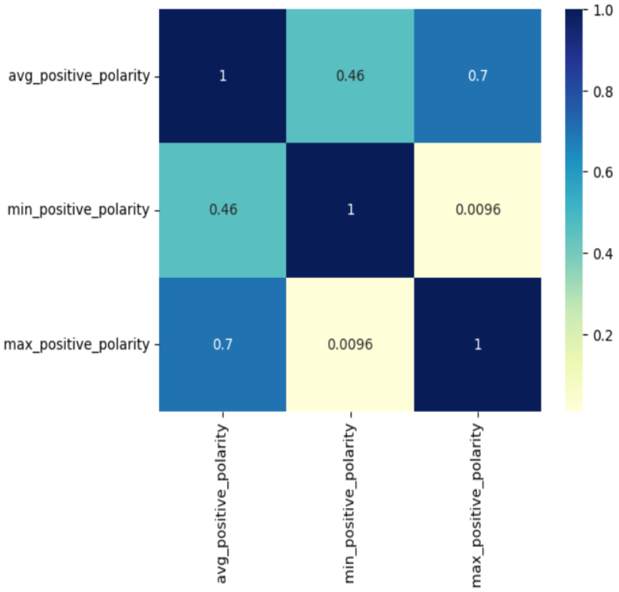

Figure 6. Correlation Map for Polarity

5.4.5 Removing/Altering attributes due to low contributions

Attributes like subjectivity and sentiment has a constant average number of shares plot for the entire range of the attributes. Hence, we either remove this attribute or we remove it from the dataset.

5.5. Standardizing the dataset

The dataset was standardised using the mean and variance of the samples, as required by the particular model.

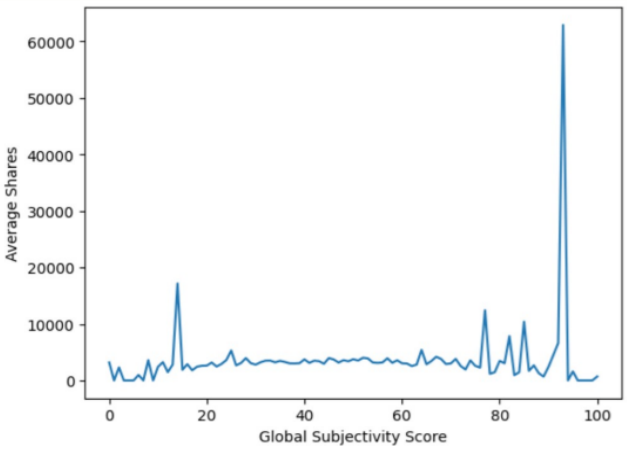

Figure 7. Average Number of Shares vs/ Global Subjectivity

6. Model Details

6.1. Random Forest Classifier

The Random Forest Classifier, an efficient machine learning algorithm, amalgamates outputs from multiple decision trees to yield a singular result – the mode of individual tree outputs. Renowned for its versatility, robustness, and superior performance across diverse datasets, it stands as a preferred choice in the realm of machine learning models. Leveraging advantages such as parallelization, resistance to overfitting, implicit feature selection, high accuracy, adaptability to missing data, resilience to outliers, and versatility, the Random Forest model emerged as the optimal solution for our dataset. Particularly noteworthy is its prowess in handling large datasets, rendering it a commendable choice for establishing a baseline in our model.

In our implementation, we employed the built-in Random Forest classifier from the scikit-learn library in Python on our preprocessed dataset. The model exhibited a maximum accuracy of **0.4** before cleaning the data. These results underscore the model's robustness and effectiveness in capturing intricate patterns within our data. After analysing the features and pre-processing the data using **Local Outlier Factor**, our model gave an accuracy of **0.98** with the test set and **0.84** as validation accuracy. Overall, our utilization of Random Forest serves as a solid foundation for further model refinement and exploration in our machine learning endeavors.

6.2. Multiclass Logistic Regression

Since our machine learning model was classification based, logistic regression seemed like the natural choice for it. Multiclass logistic regression extends the binary logistic regression to handle multiple classes. The model randomly initializes weights and biases and uses these to obtain a linear combination of the input features. This linear combination is then converted to a probability using a logistic/activation function(like sigmoid function). We then define a decision boundary, using which we obtain our output. This output is optimized by using optimization algorithms like gradient descent.

This model was the right choice for our dataset as it is able to efficiently fit on large datasets and gradient descent can be performed in batches. Being a linear model, it is simple and can be easily interpreted. It serves as a good baseline for more complex models and is also computationally efficient being a simple model. We used the built-in logistic regression in the scikit-learn library in Python. Our model gave an accuracy of **0.8** with the test set and **0.72** as validation accuracy. The low accuracy achieved in this model maybe due to the curse of dimensionality.

6.3. Gaussian Naive Bayes

Gaussian Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem. It assumes that the features in a dataset are independent of each other and that each feature follows a Gaussian (normal) distribution. The algorithm calculates the probability that a given data point belongs to a particular class by estimating the likelihood of observing its features in that class using Gaussian probability density functions. It then selects the class with the highest probability as the predicted class for the data point. Gaussian Naive Bayes is commonly used for classification tasks, especially when dealing with continuous data and assuming independence among features simplifies the modeling process.

Gaussian Naive Bayes internally performs standardization, hence it was not needed to be done. GNB models are robust to overfitting and are great models for datasets wtih large number of attributes, making it an excellent choice for our project.

We applied the SVM classifier on our dataset using the skleanr library of Python and achieved a test accuracy of **0.979** with data standardisation. After performing cross validation, we achieved a maximum accuracy of **0.884**.

6.4. Support Vector Machine(SVM)

SVM is a powerful machine learning algorithm in supervised learning that is widely used for both classification and regression tasks. It works by finding the best hyperplane that separates the given data points of different classes and maximises the margin between these points. It is a highly versatile algorithm that is capable of handling both linear and non-linear classification problems. It performs the classification task by using various kernel functions to transform the data into higher dimension spaces.

This model is a good choice for our dataset for a variety of reasons. It is effective in high dimensional spaces and since our dataset had 50 input features, it seemed only natural to try out SVM as our classification model. SVM strives to find an optimal hyperplane separating the data points meaning it results in a better decision boundary than other classification models. After performing the imperative step of standardization, We applied the SVM classifier on our dataset using the skleanr library of Python and achieved a test accuracy of **0.979** with data standardisation. After performing 10-fold cross validation, we achieved a maximum accuracy of **0.872**.

7. Results and analysis

7.1. Random Forest Classifier

Random Forest Classifier achieved an accuracy of 0.97 after standardization and 0.82 after 10-fold cross validation.

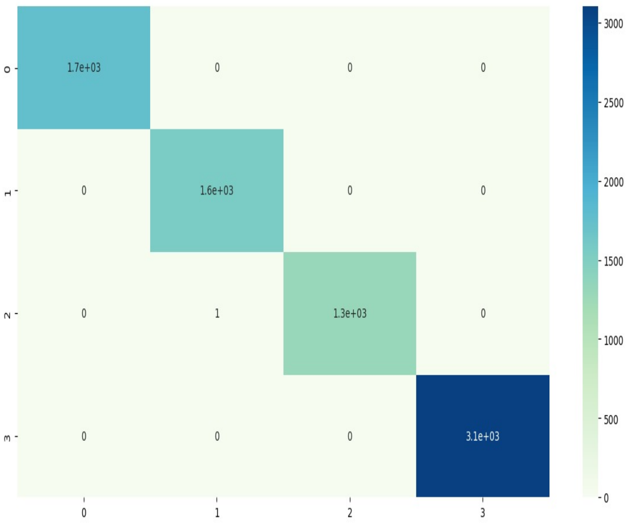

Figure 8. Confusion Matrix for Random Forest Classifier

7.2. Gaussian Naive Bayes(GNB) Classifier

Gaussian Naive Bayes Classifier achieved an accuracy of 0.89 and an accuracy of 0.89 after 10-fold cross validation. after 10-fold cross validation.

7.3. Support Vector Machine(SVM)

We experimented with different kernels and found the linear kernel to be the best. After iterating on the C parameter and using standardization we achieved an accuracy of 0.98 and an accuracy of 0.87 for 10-fold cross validation.

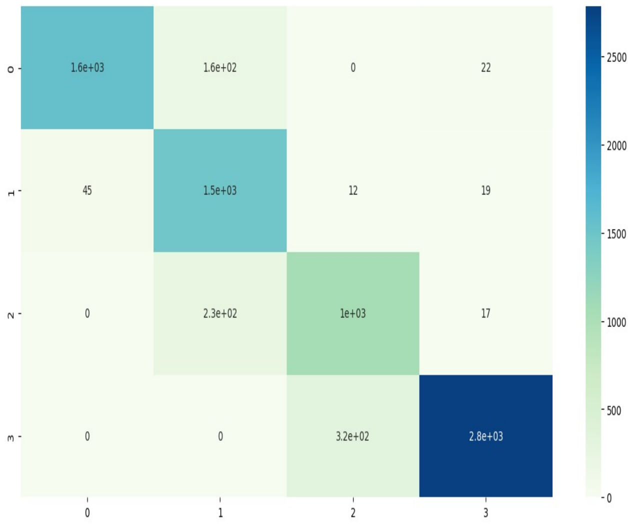

Figure 9. Confusion Matrix for Gaussian Naive Bayes Classifier

7.4. Multiclass Logistic Regression

Gaussian Naive Bayes Classifier achieved an accuracy of 0.8 and an accuracy of 0.72

8. Conclusion

Random Forest Classifier and Support Vector Machine (SVM) demonstrated exceptional accuracy, achieving approximately 0.97, whereas the Gaussian Naive Bayes (GNB) classifier achieved an accuracy of 0.89. During 10-fold cross-validation, the GNB Classifier maintained its accuracy at 0.89, whereas the other models experienced a decrease of nearly 0.1 in accuracy. This divergence can be attributed to the inherent robustness of GNB against overfitting, a quality not shared by the other models.

It's worth noting that all our models surpassed existing models when dealing with this dataset in a challenging 4-class classification problem.

9. References

[1] Fernandes,Kelwin, Vinagre,Pedro, Cortez,Paulo, and Sernadela,Pedro. (2015). Online News Popularity. UCI Machine Learning Repository. https://doi.org/10.24432/C5NS3V.

[2] Namous, Feras & Rodan, Ali & Javed, Yasir. (2018). Online News Popularity Prediction. 180-184. 10.1109/CTIT.2018.8649529.

[3] H. M. Arafat, D. H. Sagar, K. Ahmed, B. K. Paul, M. Z. Rahman and M. A. Habib, ”Popularity Prediction of Online News Item Based on Social Media Response,”

2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR), Spokane, WA, USA, 2019, pp. 173-177, doi: 10.1109/ICIEV.2019.8858525.