IS597 MLC: Final Project Report

NetID: sjain224
Student Name: Samarth Jain

Title

Predicting Online News Article Popularity: A Machine Learning Approach to Understanding Engagement Metrics

Introduction (Motivation & Objective)

The digital media landscape has undergone a significant transformation in recent years, with online news platforms becoming a primary source of information for millions of people worldwide. As the volume of content continues to grow exponentially, understanding what makes an article popular has become crucial for publishers, content creators, and marketers alike. The ability to predict an article's potential popularity before publication can significantly influence content strategy, resource allocation, and audience engagement. This project focuses on leveraging machine learning techniques to analyze and predict the popularity of online news articles based on various content and contextual features.

The reason to choose this project/dataset roots from the increasing importance of data-driven decision-making in digital publishing. By developing a robust predictive model for article popularity, the project's aim is to provide valuable insights that can help content creators optimize their articles for maximum engagement. The objective is to explore the relationships between various article attributes (such as topic, writing style, publication time, and multimedia elements) and their impact on shareability. This project will not only contribute to the understanding of content virality but also demonstrate the practical application of machine learning in the digital media industry. Through this analysis, I hope to uncover patterns and factors that significantly influence an article's potential to become widely shared, thereby enabling more informed content creation and distribution strategies.

Research Questions:

1. What are the most influential features in determining the popularity (number of shares) of an online news article?
2. How accurately can we predict the number of shares an article will receive based on its content and metadata?
3. Are there specific combinations of article attributes that consistently lead to higher shareability across different news categories?
4. How accurately can the model predict the popularity of the articles based on the number of shares?

Related Articles

1. Bandari, Roja, Sitaram Asur, and Bernardo A. Huberman. "The Pulse of News in Social Media: Forecasting Popularity." *Proceedings of the International AAAI Conference on Web and Social Media* 6, no. 1 (2012): 26-33.
    a. This article explores the factors influencing the popularity of news articles shared on social media platforms. The authors develop a forecasting model that predicts the number of shares based on various features, including social interactions and content characteristics. Their findings highlight the importance of understanding how social media dynamics affect news virality.

2. Benson, Meredith. "Predicting Online News Article Virality Using Textual Data." University of Michigan / EECS 595, 2022.
    a. This paper focuses on predicting the relative virality of online news articles based solely on the textual data in the title and headline. The author explores various approaches, including sentiment analysis and natural language processing techniques, to identify factors that contribute to an article's potential to go viral. The study emphasizes the importance of article titles in determining shareability, especially given that many users share articles without reading them.

3. Moniz, Nuno, and Luís Torgo. "Multi-Source Social Feedback of Online News Feeds." arXiv, October 17, 2018.
    a. This work presents a dataset for studying the popularity of online news articles across multiple social media platforms. The authors collected data from various news sources and social networks, providing a comprehensive view of how articles perform across different platforms. This dataset offers valuable insights for researchers studying news virality and social media dynamics.

4. Berger, Jonah, and Katherine L. Milkman. "What Makes Online Content Viral?" Journal of Marketing Research 49, no. 2 (2012): 192-205.
    a. This study examines the characteristics of New York Times articles that make them more likely to be shared. The authors analyze various factors, including emotional valence and practical utility, to understand what drives content virality. Their findings suggest that content that evokes high-arousal emotions, whether positive or negative, is more likely to be shared, providing valuable insights for predicting and understanding online content popularity.

Citations:
[1] Article 1
[2] Article 2
[3] Article 3
[4] Article 4

Data

A. Data Collection

Dataset Source - Dataset URL

The dataset I used for this project is the Online News Popularity dataset from the UCI Machine Learning Repository. It contains 39,797 instances, which exceeds the minimum requirement of 30,000 rows for effective model training. The dataset comprises 61 attributes, including 58 predictive features, 2 non-predictive features (URL and timedelta), and 1 target variable (shares).
I created 2 more variables – namely, ‘popularity’ and ‘popularity_numeric’ – to predict the popularity of the articles using different models and to find how accurately are the models predicting the popularity.
The target class for the supervised learning task is the 'shares' column, representing the number of social media shares for each article. This can be used directly for regression tasks or converted into categories for classification. The features include various aspects of the articles such as keywords, links, digital media elements, day of publication, and content analysis results. The dataset's format is structured with clear column names, making it readily accessible for analysis using standard data science libraries in Python such as pandas.

Attribute Name	Description
url	URL of the article (non-predictive)
timedelta	Days between the article publication and the dataset acquisition (non-predictive)
n_tokens_title	Number of words in the title
n_tokens_content	Number of words in the content
n_unique_tokens	Number of unique words in the content
n_non_stop_words	Number of non-stop words in the content
n_non_stop_unique_tokens	Number of unique non-stop words in the content
num_hrefs	Number of links
num_self_hrefs	Number of links to other articles published by Mashable
num_imgs	Number of images
num_videos	Number of videos
average_token_length	Average length of the words in the content
num_keywords	Number of keywords in the metadata
data_channel_is_*	Binary indicators for data channel (lifestyle, entertainment, business, social media, tech, world)
kw_max_min, kw_avg_min, kw_min_max, kw_max_max, kw_avg_max, kw_min_avg, kw_max_avg, kw_avg_avg	Aggregated metrics of keywords
weekday_is_*	Binary indicators for each day of the week
is_weekend	Binary indicator for weekend
LDA_00, LDA_01, LDA_02, LDA_03, LDA_04	Closeness to 5 LDA topics
global_subjectivity	Text subjectivity
global_sentiment_polarity	
global_rate_positive_words, global_rate_negative_words	Rate of positive and negative words in the content
rate_positive_words, rate_negative_words	Rate of positive and negative words among non-neutral tokens
avg_positive_polarity, min_positive_polarity, max_positive_polarity	Avg./min./max. polarity of positive words
avg_negative_polarity, min_negative_polarity, max_negative_polarity	Avg./min./max. polarity of negative words
title_subjectivity	Title subjectivity
title_sentiment_polarity	Title polarity

Attribute Name	Description
abs_title_subjectivity	Absolute subjectivity level
abs_title_sentiment_polarity	Absolute polarity level
shares	Number of shares (target variable)
DATAFRAME VARIABLES (created during preprocessing)	
popularity	Popularity of the article (‘popular’ or ‘not popular’)
popularity_numeric	Converting popularity variable to binary

B. Data Pre-processing

- The data pre-processing phase involves several steps to ensure the dataset is clean and ready for analysis. The CSV file is imported and an initial examination of the data is performed using functions like df.info() and df.describe() to understand the structure and basic statistics of the dataset. Handling of missing values is done using df.isnull().sum(), although the dataset is noted to have no missing values.
- Duplicate entries will be identified and removed if any exist using df.drop_duplicates(). As the names of the column had extra spaces, I used df.columns.str.strip() function to remove unnecessary spaces. To drop ‘NaN’ values from the ‘shares’ column, I used df.dropna(subset=['shares'], inplace=True) function.
- After the initial steps of data preparation, I then calculated a threshold of the number of shares using threshold = df['shares'].median() function. This helped me in assigning the articles as ‘popular’ or ‘not popular’ in the popularity variable in the dataframe, and then converted it into binary and stored it in popularity_numeric variable.
- After that, I noticed that there were a few outliers in the dataset. So, first I plotted a graph to properly visualize the outliers and then handled it using .quantile(0.99) function.

Visualizations

I made several visualizations/charts for easy understanding of the data and to help in the interpretation for feature selection and reduction.

- Plot 1 – Bar plot – Top 10 Features Correlated with Shares
    - kw_avg_avg is the feature with the strongest positive correlation with shares, suggesting that articles with higher average keyword frequencies are more likely to be shared.
    - LDA_03 is the second highest correlated feature, indicating that this LDA topic might be associated with higher share counts.
    - Other features with notable correlations include kw_max_avg, kw_min_avg, num_hrefs, num_imgs, and features related to self-references.

- Plot 2 – Scatter plot – Shares v/s Number of Tokens in Title
    - There doesn't appear to be a strong or consistent relationship between the number of tokens in a title and the number of shares. The points are scattered across the plot without a clear pattern.
    - There are clusters of points around certain numbers of tokens in the title, suggesting that some title lengths might be more common than others.
- Plot 3 – Scatter plot – Shares v/s Number of Images in the article
    - There doesn't appear to be a strong or consistent relationship between the number of images and the number of shares. The points are scattered across the plot without a clear pattern.
    - There are clusters of points around certain numbers of images, suggesting that some image counts might be more common than others.
- Plot 4 – Bar plot – Average Shares by Weekend v/s Weekday
    - The bar for "Weekend" is significantly taller than the bar for "Weekday", indicating that articles published on weekends tend to receive more shares on average.
    - Interpretation:
        - People might have more free time on weekends to browse and share articles online.
        - Social media platforms might see increased activity on weekends, leading to more shares.
        - Social media algorithms might prioritize content published on weekends.

Then, I plotted several heatmaps to find the correlation of features with 'shares' variable.

- Heatmap 1 : Content, Links and Media Features with shares
    - The features related to keyword frequencies (kw_min_min, kw_max_min, etc.) seem to have strong correlations among themselves, suggesting that these features might be redundant or highly informative.
    - There are some moderate correlations between keyword frequency features and the number of shares, indicating that keyword usage might play a role in article sharing.
    - The features related to the number of tokens, stop words, and unique tokens are highly correlated with each other, which is expected.
- Heatmap 2 : Channel and Category Features with shares
    - The strong negative correlations between various data channels and the "world" channel suggest that the "world" channel might be relatively distinct from the other channels.
    - The weak positive correlations between channels suggest that there might be some overlap between certain categories of articles.
- Heatmap 3 : Timing and Context Features with shares
    - The strong positive correlations between "is_weekend" and the individual weekend days (Saturday and Sunday) are expected and logical.
    - The LDA features (LDA_00 to LDA_04) exhibit a wide range of correlations, suggesting varying degrees of association between them.
    - Some LDA features, like LDA_04, show very weak correlations with others, indicating minimal association.
- Heatmap 4 : Sentiment and Subjectivity Features with shares
    - The strong positive correlations between rate_positive_words, rate_negative_words, and their global counterparts highlight the consistency in measuring word frequencies.
    - The strong positive correlation between global_sentiment_polarity and rate_positive_words suggests that the frequency of positive words is a significant factor in determining overall sentiment.

    - Many of the correlations between sentiment-related features and word frequencies are weak, indicating that the relationship between these factors might be complex or influenced by other variables.
- Heatmap 5 : Self-reference Features with shares
    - The strong positive correlations between the different self-reference metrics (min, max, avg) suggest that these features are highly related to each other.
    - The very weak correlations between shares and the self-reference metrics indicate that the number of self-references might not be a strong predictor of the number of shares.

Data Preparation and Preprocessing for Modeling

After the initial data cleaning and correlation analysis, the next step involved preparing the data for machine learning models. Here's a detailed explanation of the steps taken to define features, split the data, scale the features, and apply dimensionality reduction.

- Defining Features and Target Variables
    - To set up the data for both classification and regression tasks, the features and target variables were defined as follows:
    - Features (X): All columns except url, timedelta, shares, popularity, and popularity_numeric were selected as features. This is because url and timedelta are not relevant for the predictive tasks, and shares and popularity_numeric are the target variables.
    - Classification Target (y_classification): The popularity_numeric column was used as the target variable for the classification task. This column categorizes the popularity of news articles into different levels.
    - Regression Target (y_regression): The shares column was used as the target variable for the regression task, predicting the actual number of shares.
- Splitting Data into Training and Testing Sets
    - The dataset was split into training and testing sets to evaluate the performance of the models. This was done using the train_test_split function from Scikit-learn.
    - Training and Testing Sets: The data was split into training and testing sets with a test size of 20% (0.2). This ensures that the models are trained on 80% of the data and evaluated on the remaining 20%.
    - Random State: The random_state parameter was set to 42 to ensure reproducibility of the splits.
- Scaling Numeric Features
    - To ensure that all features are on the same scale, which is important for many machine learning algorithms, the numerical features were scaled using the StandardScaler from Scikit-learn.
    - StandardScaler: This scaler standardizes features by removing the mean and scaling to unit variance.
    - Fit and Transform: The fit_transform method was used on the training data to fit the scaler and transform the data. The transform method was used on the testing data to apply the same scaling.
- Dimension Reduction using PCA
    - To reduce the dimensionality of the feature space and potentially improve model performance, Principal Component Analysis (PCA) was applied.
    - PCA: The PCA class from Scikit-learn was used with n_components=0.95, meaning that the number of components was chosen such that 95% of the variance in the data is retained.

- Fit and Transform: The fit_transform method was used on the training data to fit the PCA and transform the data. The transform method was used on the testing data to apply the same transformation.

Analysis & Methodology

Classification Models

To predict the popularity of news articles, I evaluated several classification models. Here's a detailed description of the models used, and the methodology followed:

Model Selection

- Logistic Regression: A baseline model for classification tasks, known for its simplicity and interpretability.
- Random Forest Classifier: An ensemble model that combines multiple decision trees to improve accuracy and reduce overfitting.
- Gradient Boosting Classifier: Another ensemble model that uses a series of weak models to create a strong predictive model.
- Support Vector Classifier (SVC): A model that finds the hyperplane that maximally separates the classes in the feature space.
- Multi-Layer Perceptron (MLP) Classifier: A neural network model that can learn complex relationships between features.

Model Evaluation

- Accuracy: The proportion of correctly classified instances.
- Precision: The ratio of true positives to the sum of true positives and false positives.
- Recall: The ratio of true positives to the sum of true positives and false negatives.
- F1-score: The harmonic mean of precision and recall.
- ROC-AUC: The area under the receiver operating characteristic curve, which measures the model's ability to distinguish between classes.

The predictions were made on the test set, and the metrics were calculated using the actual labels and predicted labels.

Regression Models

To predict the number of shares for news articles, I evaluated several regression models. Here's a detailed description of the models used, and the methodology followed:

Model Selection

The following regression models were chosen for their ability to handle continuous target variables and their robust performance in various scenarios:

- Linear Regression: A baseline model for regression tasks, known for its simplicity and interpretability.
- Logistic Regression: Although primarily a classification model, it was included to compare its performance in a regression context.
- Random Forest Regressor: An ensemble model that combines multiple decision trees to improve accuracy and reduce overfitting.
- Gradient Boosting Regressor: Another ensemble model that uses a series of weak models to create a strong predictive model.

- Support Vector Regressor (SVR): A model that finds the hyperplane that minimizes the error in the feature space.
- Multi-Layer Perceptron (MLP) Regressor: A neural network model that can learn complex relationships between features.

Model Evaluation

- Mean Absolute Error (MAE): The average difference between predicted and actual values.
- Root Mean Squared Error (RMSE): The square root of the mean of the squared differences between predicted and actual values.
- R^2 Score: Measures the goodness of fit of the model, with higher values indicating better fit.

The predictions were made on the test set, and the metrics were calculated using the actual labels and predicted labels.

Results

Confusion Matrices and Count Plots

- For Logistic Regression :
  - True Positives (TP): 2935 instances were correctly predicted as Popular.
  - True Negatives (TN): 2148 instances were correctly predicted as Not Popular.
  - False Positives (FP): 1510 instances were incorrectly predicted as Popular (Type I error).
  - False Negatives (FN): 1257 instances were incorrectly predicted as Not Popular (Type II error).
- For Random Forest :
  - True Positives (TP): 2928 instances were correctly predicted as Popular.
  - True Negatives (TN): 2143 instances were correctly predicted as Not Popular.
  - False Positives (FP): 1515 instances were incorrectly predicted as Popular (Type I error).
  - False Negatives (FN): 1264 instances were incorrectly predicted as Not Popular (Type II error).
- For Gradient Boosting :
  - True Positives (TP): 2998 instances were correctly predicted as Popular.
  - True Negatives (TN): 2082 instances were correctly predicted as Not Popular.
  - False Positives (FP): 1576 instances were incorrectly predicted as Popular (Type I error).
  - False Negatives (FN): 1194 instances were incorrectly predicted as Not Popular (Type II error).
- For Support Vector Classifier (SVC) :
  - True Positives (TP): 2974 instances were correctly predicted as Popular.
  - True Negatives (TN): 2194 instances were correctly predicted as Not Popular.
  - False Positives (FP): 1464 instances were incorrectly predicted as Popular (Type I error).
  - False Negatives (FN): 1218 instances were incorrectly predicted as Not Popular (Type II error).
- For Multi-Layer Perceptron (MLP) :
  - True Positives (TP): 2735 instances were correctly predicted as Popular.
  - True Negatives (TN): 2258 instances were correctly predicted as Not Popular.
  - False Positives (FP): 1400 instances were incorrectly predicted as Popular (Type I error).
  - False Negatives (FN): 1457 instances were incorrectly predicted as Not Popular (Type II error).

Comparison

Model	True Positive	True Negative	False Positive	False Negative
Logistic Regression	2935	2148	1510	1257
Random Forest	2928	2143	1515	1264
Gradient Boosting	2998	2082	1576	1194
SVC	2974	2194	1464	1218
MLP	2735	2258	1400	1457

Observations

- True Positives (TP): Gradient Boosting has the highest TP, followed closely by SVC. MLP has the lowest TP.
- True Negatives (TN): MLP has the highest TN, followed by SVC and Logistic Regression. Gradient Boosting has the lowest TN.
- False Positives (FP): MLP has the lowest FP, followed by SVC. Gradient Boosting has the highest FP.
- False Negatives (FN): Gradient Boosting has the lowest FN, followed by SVC. MLP has the highest FN.

ROC-AUC

- Logistic Regression
  - The ROC curve for Logistic Regression indicates that the model has a moderate performance in distinguishing between positive and negative classes. The AUC of 0.70 suggests that the model is better than random guessing, but there's still room for improvement.
- Random Forest
  - The ROC curve for Random Forest shows that it has a moderate performance distinguishing between positive and negative classes, like the Logistic Regression model. The AUC of 0.70 indicates that both models are better than random guessing, but there's still room for improvement.
- Gradient Boosting
  - The ROC curve for Gradient Boosting shows that it has a moderate performance in distinguishing between positive and negative classes, with a slight improvement over Logistic Regression and Random Forest based on the AUC.
- SVC
  - The ROC curve for SVC shows that it has a good performance in distinguishing between positive and negative classes, with the highest AUC among all the models analyzed.
- MLP
  - The ROC curve for MLP shows that it has a moderate performance in distinguishing between positive and negative classes, with a slightly lower AUC compared to some of the other models.

Comparison

Model	AUC
SVC	0.72
Gradient Boosting	0.71
Logistic Regression	0.70
Random Forest	0.70
MLP	0.68

Observations

- SVC has the highest AUC, indicating the best overall performance in distinguishing between positive and negative classes.
- Gradient Boosting has the second-highest AUC, suggesting slightly better performance than Logistic Regression and Random Forest.
- Logistic Regression and Random Forest have similar AUC values, indicating comparable performance.
- MLP has the lowest AUC, suggesting potentially lower performance compared to the other models.

Classification Models

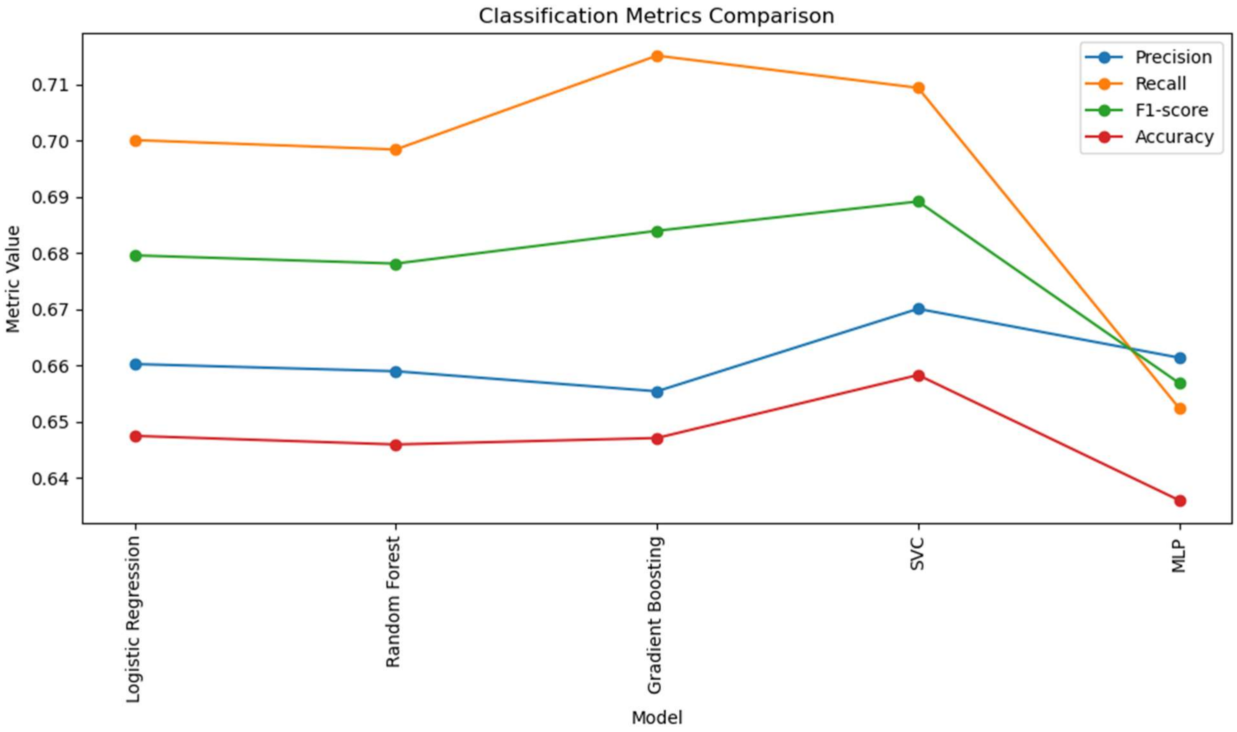
Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Logistic Regression	0.6475	0.6603	0.7001	0.6796	0.6970
Random Forest	0.6460	0.6590	0.6985	0.6782	0.7028
Gradient Boosting	0.6471	0.6554	0.7152	0.6840	0.7053
SVC	**0.6583**	**0.6701**	**0.7094**	**0.6892**	**0.7168**
MLP	0.6361	0.6614	0.6524	0.6569	0.6839

Insights and Observations

- The SVC model consistently demonstrates the best performance across all metrics (accuracy, precision, recall, F1-score, and ROC-AUC). This suggests that SVC is well-suited for this task of classifying news article popularity.
- Gradient Boosting and Random Forest, both ensemble methods, show competitive performance, highlighting their effectiveness in handling complex datasets and potentially capturing subtle patterns in the data.
- Logistic Regression, a simpler model, provides reasonable performance metrics, serving as a good baseline for comparison.
- MLP, while a powerful model, shows the lowest performance in this case. This could potentially be due to factors like:
  - Overfitting: The MLP model might be overfitting the training data, leading to poor generalization on the test set.
  - Hyperparameter Tuning: The MLP model might require more careful tuning of its hyperparameters (e.g., number of layers, number of neurons, learning rate) to achieve better performance.

- The ROC-AUC scores provide a valuable comparison of the models' ability to distinguish between positive and negative classes, independent of the classification threshold. SVC demonstrates the highest ROC-AUC, further supporting its strong performance.

Overall, these results suggest that the choice of classification model can significantly impact the accuracy of predicting news article popularity. SVC appears to be the most promising model based on the current analysis, but further investigation and tuning might reveal even better performance from other models.



Regression Models

Model	MAE	RMSE	R-Squared Score
Linear Regression	2026.21	3585.12	0.0596
Logistic Regression	1908.48	4259.68	-0.3276
Random Forest	2165.34	3652.25	0.0240
Gradient Boosting	2027.41	3587.32	0.0584
SVR	1751.10	3893.14	-0.1090
MLP Regressor	1993.81	3559.79	0.0728

Insights and Observations:

- SVR achieved the lowest Mean Absolute Error (MAE) of 1751.10, indicating that it generally made the smallest average prediction errors.
- Most models, including SVR, had negative R^2 scores. This suggests that the models did not fit the data well and their predictions were not significantly better than simply predicting the mean number of shares.
- Model Performance:
  - SVR: Lowest MAE but negative R^2 score indicates potential overfitting or difficulty capturing the underlying relationship.

  - Linear Regression, Gradient Boosting, and MLP Regressor: Showed relatively similar performance with moderate MAE and low R² scores, suggesting that these models may not be well-suited for this regression task.
  - Logistic Regression and Random Forest Regressor: Had lower performance in terms of MAE and R² score, indicating a poorer fit to the data compared to other models.
- Predicting the exact number of shares for an article is likely a challenging task due to the complex and multifaceted factors influencing popularity.

Overall, the results suggest that predicting the exact number of shares using regression models is challenging in this dataset. While SVR achieved the lowest MAE, it also had a negative R² score, indicating room for improvement. Further exploration and model tuning are necessary to achieve better predictive performance.

Discussion and Conclusion

Classification Performance:

- SVC emerged as the top-performing model for classification, demonstrating the best performance across accuracy, precision, recall, F1-score, and ROC-AUC.
- Gradient Boosting and Random Forest, both ensemble methods, exhibited competitive performance, highlighting their ability to capture complex relationships within the data.
- Logistic Regression, despite its simplicity, provided a solid baseline, demonstrating its effectiveness as a starting point for classification tasks.
- MLP showed the lowest performance, which could be attributed to factors like overfitting or the need for more extensive hyperparameter tuning.

Regression Performance:

- Predicting the exact number of shares proved to be a challenging task for all regression models.
- SVR achieved the lowest MAE, suggesting it might be better at making smaller prediction errors. However, the negative R² scores for most models, including SVR, indicate that they did not fit the data well and their predictions were not significantly better than simply predicting the mean number of shares.
- This suggests that predicting the exact number of shares might require more complex models or a deeper understanding of the underlying factors influencing popularity.

ROC-AUC Analysis:

- The ROC-AUC scores provided valuable insights into the models' ability to distinguish between popular and not popular articles.
- SVC demonstrated the highest ROC-AUC, further supporting its strong performance in classification.

I explored the performance of various classification and regression models in predicting news article popularity. The results demonstrate that:

- Classification: SVC appears to be the most promising model for classifying news articles as popular or not popular based on its superior performance across multiple metrics.
- Regression: Predicting the exact number of shares proved to be challenging for all regression models. Further exploration and model tuning are necessary to improve predictive performance.

Future Work

1. **Feature Engineering and Selection:**
   a. Investigate new features: Explore and engineer new features that could potentially improve model performance. This could include:
   b. Temporal features: Incorporate time-based features such as day of the week, time of day, and seasonality.
   c. Feature selection techniques: Employ feature selection techniques (e.g., recursive feature elimination, feature importance analysis) to identify the most informative features and potentially reduce model complexity.
2. **Model Enhancement:**
   a. Hyperparameter tuning: Conduct more extensive hyperparameter tuning for each model using techniques like grid search, random search, or Bayesian optimization.
   b. Ensemble methods: Explore advanced ensemble methods, such as stacking or blending, to combine the strengths of different models.
3. **Addressing Class Imbalance:**
   a. Resampling techniques: If the dataset exhibits class imbalance (e.g., more non-popular articles than popular ones), employ techniques like oversampling, under-sampling, or using weight loss functions to address the imbalance and improve model performance.
4. **Explainable AI (XAI) Techniques:**
   a. Investigate XAI techniques: Apply XAI techniques (e.g., SHAP, LIME) to understand the factors that drive model predictions and identify the most influential features. This will improve model interpretability and trust.
5. **Real-time Predictions:**
   a. Develop a real-time prediction system: Build a system that can predict article popularity in real-time, allowing for timely interventions and content adjustments.

GitHub Repository

Repository Link – GitHub Link

References

1. Arapakis, I., Lalmas, M., Cambazoglu, B. B., Marcos, M. C., & Jose, J. M. (2014). User engagement in online News: Under the scope of sentiment, interest, affect, and gaze. Journal of the Association for Information Science and Technology, 65(10), 1988-2005.
2. Bandari, R., Asur, S., & Huberman, B. A. (2012). The Pulse of News in Social Media: Forecasting Popularity. Proceedings of the International AAAI Conference on Web and Social Media, 6(1), 26-33.
3. Fernandes, K., Vinagre, P., & Cortez, P. (2015). A Proactive Intelligent Decision Support System for Predicting the Popularity of Online News. In F. Pereira, P. Machado, E. Costa, & A. Cardoso (Eds.), Progress in Artificial Intelligence (pp. 535-546). Springer International Publishing.
4. Tatar, A., Antoniadis, P., de Amorim, M. D., & Fdida, S. (2014). From popularity prediction to ranking online news. Social Network Analysis and Mining, 4(1), 174.