Characteristics of viral news on Mashable

Albert Lee, Yueh-Chang Kuo, Conor Walsh, Jaeger Wells
MSDS 422
Winter 2024
Group Project Final Presentation 03/10/2024

Executive Summary

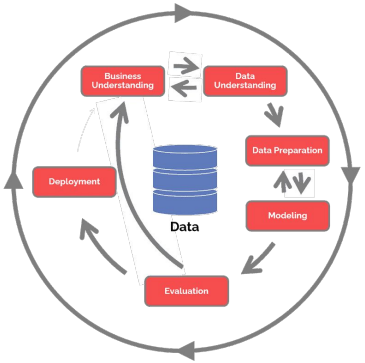
Online news and social media have become a dominant force in our society today. "Going viral" is how people learn about world events. While news feels like it is unbalanced and incredibly negative most of the time, there is a need to explore what are the attributes of articles that drive virality. By utilizing data from over 2 years from the website Mashable, we will look to understand what are the characteristics of a news article going viral.

Data Source: *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.*

Problem Statement & Process Model

Problem Statement: What are the characteristics that make news go viral? We will look to understand if subjectivity, overall sentiment, channel news is being displayed, and type of file format for news impacts the total number of shares on Mashable. Ultimately, we view this analysis as a binary classification problem: will the news be popular or unpopular?

Process Model: We have implemented a Cross-Industry Standard Process for Data Mining (CRISP-DM) as our approach to answering the problem statement. This allows us the flexibility to approach the problem either by vertical or horizontal slicing of our features.



Data Source: *https://www.datascience-pm.com/crisp-dm-2/*

Tools & Deployment

Model Development

We utilized cloud environments like **Jupyter Notebooks** to collaborate on the analysis of the data and local environments for analysis utilizing CPUs.

Github was utilized for version control and documentation.

Select Deployment Environment

It is important to select a deployment environment best suited to the needs of the project and other requirements. This could be utilizing environments such as AWS, Azure, or Google Cloud.

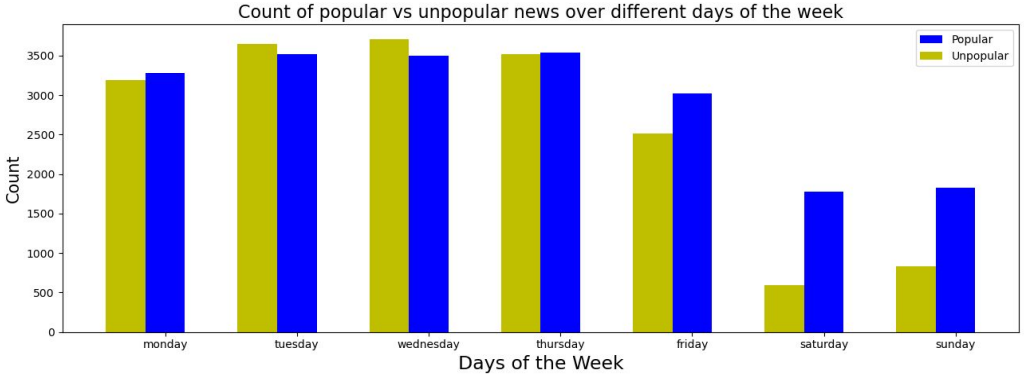
Containerization

Packaging a model, runtime environment, and its dependencies allows for scalable reproducibility as new data comes in.

Deploy, Monitor, Maintain

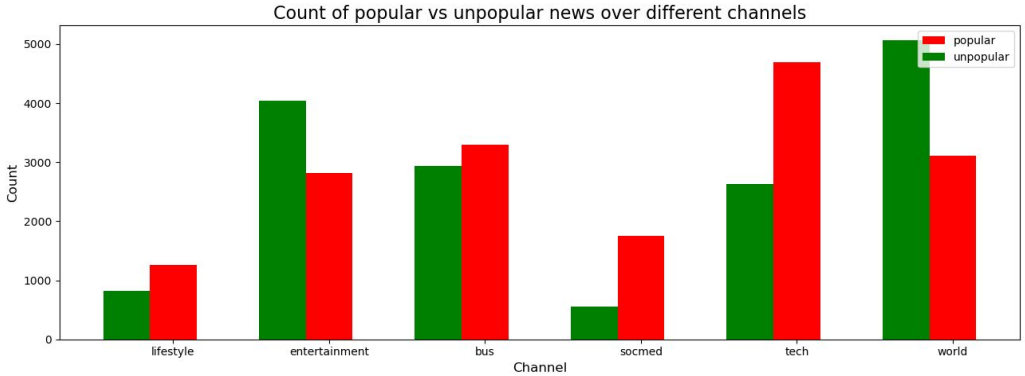
Once a model is containerized, the model is deployed and monitored for performance + to see if there are any issues that need to be taken care of with maintenance.

Exploratory Data Analysis (EDA)



Using EDA, we're able to determine the count of popular vs. unpopular news over different days of the week. News is more popular Monday, Thursday, Friday, Saturday, and Sunday. Whereas the news is more unpopular on Tuesday and Wednesday.

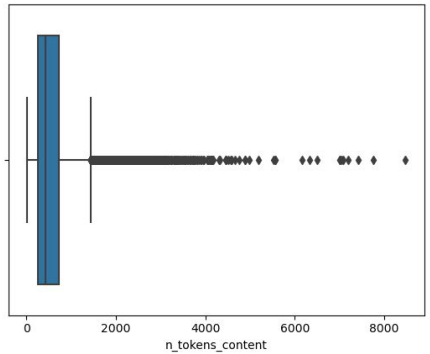
Using EDA, we're also able to determine the count of popular vs. unpopular over different channels. Lifestyle, Business, Social Media, and Tech are more popular vs. Entertainment and World are more unpopular.

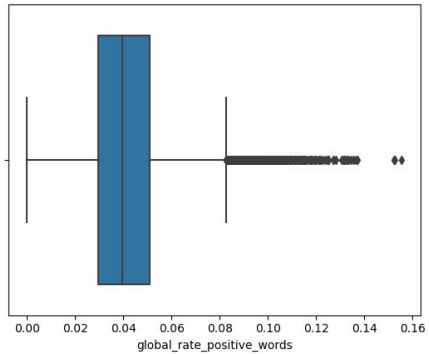


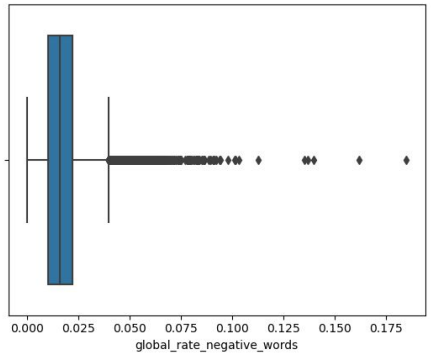
Feature Engineering - Initial DataFrame (DF)

While performing EDA, we realized that there were no strong correlations with Shares and also the dataset had a large number of outliers across the variables (see below examples). Were were able to treat the outliers using “Yeo-Johnson” methodology.

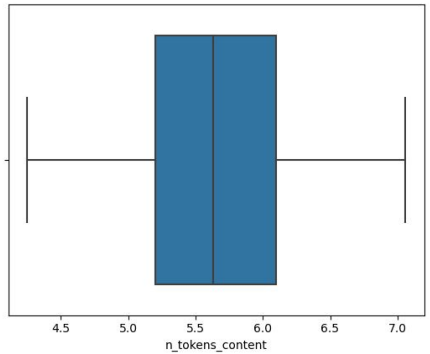
Before Feature

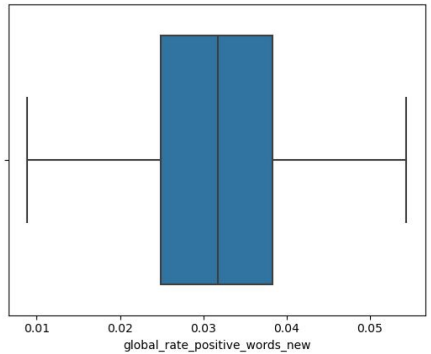


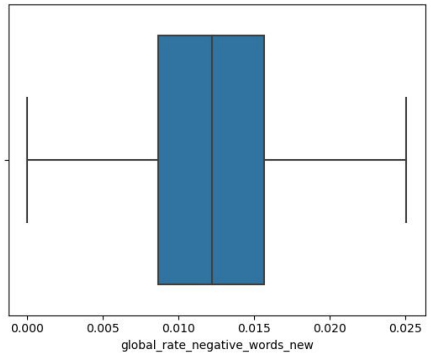




After Feature







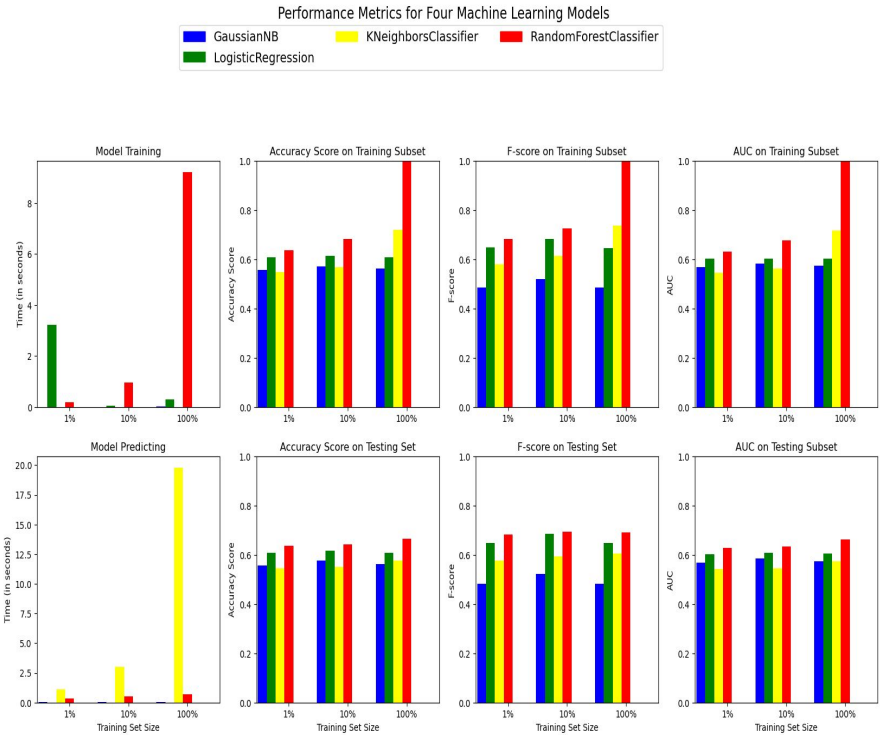
Methodology - Model Selection

Since we are approaching this as a binary classification problem, we have selected four classification algorithms to test:

1. GaussianNB: The term "Naive" is attributed to its assumption of feature independence, signifying that the presence or absence of one feature is considered unrelated to the presence or absence of another.
2. LogisticRegression: The algorithm is based on the logistic function, which transforms a linear combination of input features into a range of values between 0 and 1.
3. KNeighborsClassifier: It belongs to the category of instance-based learning, in which predictions are determined by the majority class for classification.
4. RandomForestClassifier: A classification algorithm that falls under the ensemble learning category. During training, it constructs a set of decision trees, with each tree being independently trained on a randomized subset of features and data through the utilization of bootstrapped sampling.

Methodology - Evaluation

1. Training Set Size: The performance metrics are evaluated at different training set sizes: **1%**, **10%**, and **100%**.
2. Model Training and Prediction Time: GaussianNB has the shortest training time, while RandomForestClassifier takes significantly longer. For prediction time, KNeighborsClassifier takes the longest, especially as the dataset size increases.
3. Accuracy Score, F-score, and AUC: RandomForestClassifier generally shows high performance across all metrics, followed by Logistic Regression. The strength of RandomForestClassifier appears to be more significant in the training subset.



Findings

Using Binary Classification to binarize the data into popular vs. unpopular news categories. The median threshold was 1,400 shares. From there we performed 4 models as discussed in slide 7. Using Accuracy as the performance metric (as seen in the below confusion matrices) we were able to determine that the Random Forest model performed the best across all sample datasets. We then hyper tuned the model to increase the overall accuracy up to ~67%.

	Actual	
Predicted	TP	FP
	FN	TN

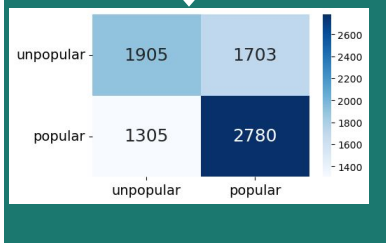
A confusion matrix is used as an evaluation tool in machine learning models.

56.2%
Gaussian NB
100% of Sample

- 1% of Sample: 55.5%
- 10% of Sample: 57.6%

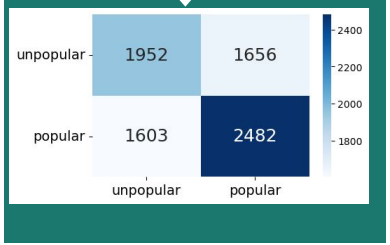
60.9%
Logistic Regression
100% of Sample

- 1% of Sample: 60.7%
- 10% of Sample: 61.7%



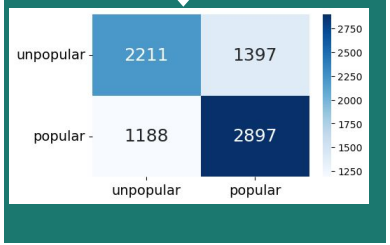
57.6%
K Neighbors
100% of Sample

- 1% of Sample: 54.4%
- 10% of Sample: 54.9%%



66.3%
Random Forest
100% of Sample

- 1% of Sample: 63.4%
- 10% of Sample: 64.1%



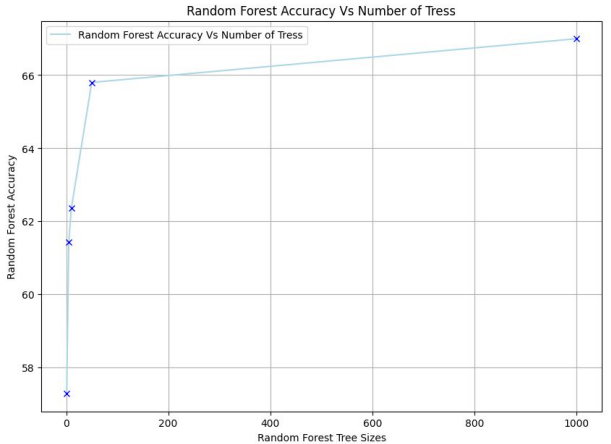
Conclusions

While we were able to predict with ~67% accuracy there is room to continue to improve the models. The random forest model classification before hyper tuning was 66.39% and after hyper tuning increased to 66.99%. With a 67% accuracy, the business should consider using these models to drive increased shares. Surprisingly, our accuracy was also in line with the original authors in their paper - “A Proactive Intelligent Decision Support System for Predicting the Popularity of Online News”.

The 67% accuracy highlights variability with predictions also depends on the dataset. The imbalance in the class distribution caused the model to be more biased towards popular articles.

As we reflect, on our analysis, we could have included other variables such as word count or certain keywords and factor that into the performance of the models.

Another interesting data that we would have liked to seen from the dataset is around variables on the author, author experience, number of articles that authors have published, etc. One possibility is that certain authors may perform better than others.



The Random Forest accuracy also increases with increasing number of random forest tree sizes.

Recommendations

Immediate Deployment of HyperTuned RF Classifier

- Algorithm capable of predicting ‘viral’ article 2 of every 3 attempts
- Additional 3 models should be maintained in the MLOps deployment pipeline to reassess performance with changing market conditions

Explore Training on Smaller Percentages of Dataset

- Our analysis indicates <3% change in performance across all models when training with 10% and 100% of the dataset
- To save computation time, training excursions may be run on smaller percentages of the dataset to determine a baseline model feasibility

Future Exploration of Multi-Label Classification to Increase Accuracy

- By exploring multi-label classification models, it is possible to continue to fine tune model accuracy, especially by content channel
- This could help provide each content area with more specific ways to supercharge user engagement.

Lessons Learned

Classification Algorithms are a Feasible Solution to Predicting if an Article will 'Go Viral'

- The HyperTuned RF Model achieved 67% accuracy, all 4 models achieved accuracy of >55%
  - Potential for Improvement with Additional Data

Outliers in Dataset Can Be Limited through Feature Engineering

- Limiting outliers in dataset limits risk of overfitting to training data and allows for higher performance on unseen test data

Popularity Driven By Weekend Viewership

- Saturday & Sunday datasets include higher percentages of popular articles

Working Through Imbalanced Dataset

- With the dataset heavily skewed above the median threshold for going viral, we should explore adding additional data sources or applying SMOTE techniques to normalize distribution via oversampling with synthetic data.

Thank You

Please reach out to our team for model maintenance support!

Albert Lee
- albertlee2025@u.northwestern.edu

Jaeger Wells
- jaegerwells2024@u.northwestern.edu

Yueh-Chang Kuo
- yueh-changkuo2025@u.northwestern.edu

Conor Walsh
- conor.walsh2026@u.northwestern.edu

References

Logunva, I. (2023). ML model deployment: Challenges, solutions & best practices. Retrieved from https://serokell.io/blog/ml-model-deployment

Holtz, N. (2023). What is CRISP DM?. Data Science Process Alliance. Retrieved from https://www.datascience-pm.com/crisp-dm-2/

Hvitfeldt, E. (2024). Numeric Transformation using Yeo-Johnson Transformation. In Feature Engineering A-Z. Retrieved from https://feaz-book.com/numeric-yeojohnson

Johnson, A., & Weinberger, D. (n.d.). Predicting News Sharing on Social Media. Stanford CS229 Project Reports. Retrieved from https://cs229.stanford.edu/proj2016/report/JohnsonWeinberger-PredictingNewsSharing-report.pdf

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. Retrieved from https://archive.ics.uci.edu/dataset/332/online+news+popularity

Jain, S. (207). Introduction to Multi-label Classification. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2017/08/introduction-to-multi-label-classification/