Predicting Popularity using Ukraine Conflict Tweets

GROUP 10 - SAS:

Stephanie Chen (schen176)
Anushree Vilas Pimpalkar (avp4)
Shantanu Solanki (solanki7)

***Abstract*—**

Russia's invasion of Ukraine in February 2022 was a major point of escalation in the Russo-Ukrainian War. This is currently the largest war in Europe since World War II and has been a main topic in the news. Many people have flooded the Internet with social media posts regarding this conflict, and some of these posts have gone viral. This begs the question of what type of posts are more prone to being popular. In this paper, we focus on analyzing public Twitter posts and responses across the world regarding this conflict to predict the degree of virality of a tweet. Specifically, we performed classification models such as Logistic Regression, Decision Trees and Random Forest to predict fixed retweet count bins/groups, which were created specifically to have the problem as a classification problem. The classification models were compared via accuracy and F1 score. We also performed regression models such as Decision Trees and Random Forest regressor to predict the retweet count. The regression models were compared via root mean squared error (RMSE) and mean absolute error (MAE). We used the text feature, tweet texts, to predict popularity. This feature was further processed by performing the following natural language processing techniques: stemming, bag-of-words, and term frequency-inverse document frequency (TF-IDF). The results indicated that for the classification problem, the logistic regression model performed the best in predicting the tweet popularity with the testing accuracy of 87.01%, and Decision Trees performed better in the regression setting.

***Keywords*—popularity prediction, classification, regression, social media, text data, RMSE, MAE, F1 Score, accuracy**

I. Introduction

On February 24, 2022, Russia invaded Ukraine, which marked a major escalation in the Russo-Ukrainian War, that has been ongoing since 2014. The invasion is internationally considered an act of aggression and has become Europe's largest refugee crisis since World War II. This crisis has had repercussions across the world. Such crises can be heavily influenced through the mobilization of public opinion on popular social media platforms, such as Twitter, Facebook, and TikTok. Numerous people have gravitated to these sites to create posts expressing their opinions and to provide insights regarding the conflict, which have generated many likes, shares, and comments. As such, these sites have gathered enormous amounts of public data that can provide valuable information.

The main goal of our study is to focus on predicting the popularity of an English tweet from Twitter regarding the conflict. We will use the text of the tweet as our feature. The popularity will be measured through the text data or content in a tweet, which plays a significant role in spreading a message's awareness across mass amounts of people. Regression models such as Decision Tree and Random Forest and classification models such as Logistic regression, Decision Tree and Random Forest will be performed to determine which is the best model to predict popularity. For regression, the retweet count will remain unchanged, but for classification, the retweet count will be converted to five bins (i.e., categories like "0-100" retweets). The regression models will be compared via their root mean squared error and mean absolute error, and the classification models will be compared via their accuracy and F1 score.

To process the text data, we will perform a natural language processing technique called stemming to reduce words to their root forms. We will also remove symbols, punctuation marks, and English stop words, which are a set of the most used English words, to simplify the texts. Finally, the texts will be converted into vectors using the natural language processing techniques called bag-of-words and TF-IDF methods. By predicting the virality of a tweet in the Twitterverse regarding this ongoing conflict, we can better understand the content that people are most drawn to, which can be impactful in influencing public opinion.

II. Related Work

Previous studies predicting online news popularity have been conducted. In this paper [1], various machine learning algorithms, such as random forest, adaptive boosting, support vector machine (SVM), K-nearest neighbor, naïve bayes, linear regression, and logistic regression are discussed, and then implemented on an article dataset provided by the UCI machine learning repository. The algorithms classified the articles as either popular or non-popular. The authors concluded that the random forest classification method performed the best because it resulted in the highest accuracy. Similarly, [2] utilized linear regression, logistic regression, SVM, and random forest to classify articles from Mashable, a well-known online news website. The results also showed that random forest produced the highest accuracy and recall. For our study, we plan to implement linear regression, logistic regression, decision trees and random forest models.

[3] used movie text regression to predict a movie's opening weekend revenue. More specifically, they performed sentiment analysis using the following text features: (1) n-grams, (2) part-of-speech n-grams, and (3) dependency relations, on movie reviews and relied on movie metadata to predict a future quantity. Linear regression was performed, and the movie revenue predictions were evaluated using mean absolute error (MAE) and Pearson's correlation (r) between the actual and

predicted revenue. The results indicated that a combination of the text and meta features achieved the best performance in terms of MAE and r, and the text features can be substituted in for and improved over other metadata features. The principle used here is the same where text data is used to predict a continuous quantitative response. Our preliminary models deal with classification, but we also plan to implement regression models too.

[4] investigated Twitter hashtags with the goal of predicting when the popularity reaches its peak. In this paper, they included three research aspects. The first aspect is examining how early popularity reaches its peak. They found that this happens in the early stage of its evolution. Second, they discussed when the peak time prediction should be triggered. Lastly, they designed a multi-modal based deep learning method, such as multi-modal embedding and attention mechanisms. They evaluated the overall performance with the minimum, quartiles, and maximum values of the absolute errors. They found that their prediction method outperformed baseline methods. Similarly, in our project, we have predicted the popularity of the tweet with respect to the retweet count. We created five bins within retweet count to classify the range that any new tweet could be in. This made our problem a multiclassification problem.

This study [5] determined which features were the most impactful in affecting the number of retweets. They observed content and contextual features from a large set of tweets to identify the features that are significantly associated with retweet rate. Then, they built a predictive model to predict the number of retweets. The results indicated that URLs, hashtags, number of followers and followees, and age of the account all had a strong association with retweet count. Our study only focuses on how impactful the text feature is in predicting retweet count.

In [6], author and content features were analyzed using logistic regression to predict whether a Polish Facebook post was considered “popular” or “unpopular”. They implemented the following two analyses: (1) compared performance of topic features using topic modeling against author features and (2) compared topic features against a bag-of-words feature set. The results from the first analysis indicated that author features performed better at predicting popularity than topic features, and the results from the second analysis also indicated that bag-of-words features performed better at predicting popularity than topic features. However, after conducting an analysis on the weighted individual features, they found that topic features were more interpretable than an ordered list of individual words. We plan to use a content feature (i.e., tweet text) to predict popularity of Twitter posts, and we will be focusing on English posts only.

Like our analysis, [7] identified the characteristics of news propagation from news media using the Twitter data. They built a news popularity prediction model to quickly predict the final number of retweets of a news tweet. Through the trace-driven experiments, they validated their model by comparing predicted popularity and real popularity and showed its superior performance in comparison with the regression prediction model. They found that the average interaction frequency between the retweeters and the news source is correlated with news popularity. They also found that, although there is some correlation between the negative sentiment of news and retweet popularity, there is no such obvious correlation with the positive sentiment of news.

Resembling closer to our work, [8] employed different machine learning classifiers like SVM, Naïve Bayes, Logistic Regression, Random Forest, and Neural Network, on top of two different text processing approaches used in NLP (natural language processing), namely bag-of-words (TFIDF) and word embeddings (Doc2Vec), to check how many likes and retweets a tweet can generate. The results obtained indicate that all the models performed 10-15% better with the bag-of-words technique. We have employed the same approach with preferring bag-of-words over the in-general more sturdy technique of Doc2Vec. We have also binned the count of retweets to categories based on their numbers, which was also done in the paper. However, because we are dealing with data that has been generating huge followings and large outbursts of public opinion, we made the bins bigger than those used in the paper.

III. Data

The dataset we are using is the Ukraine Conflict Twitter Dataset provided on Kaggle. It is primarily composed of text data, *text* (tweet text), with some quantitative features such as *following* (the number of accounts the user is following), *followers* (the number of followers the user has), and *total tweets* (the total number of tweets posted by the user) [9]. We noticed no specific trend in the *following* and *followers* in our data. Therefore, for our analysis, we will only be using *text* as a feature. The full dataset has a size close to 4 Gigabytes, with 17 features and ~18 million records. The text variable in the dataset is comprised of entries in all the prominent languages of the world. However, we will be focusing on the first 100k English tweets only, which, though, simplifies the problem, provides a blueprint for expanding our analysis to multiple languages with slight tweaks in our approach. All data cleaning and data processing was completed in PySpark. Additionally, observations with missing data were excluded from our analysis. This left us with 99,944 observations remaining in the dataset. Below, we provide a visualization of a subset of the data:

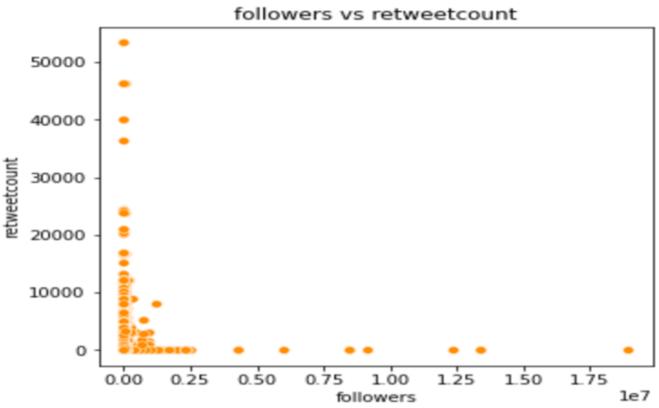


Fig. 1. Retweet Count vs no. of Followers (10 million units)

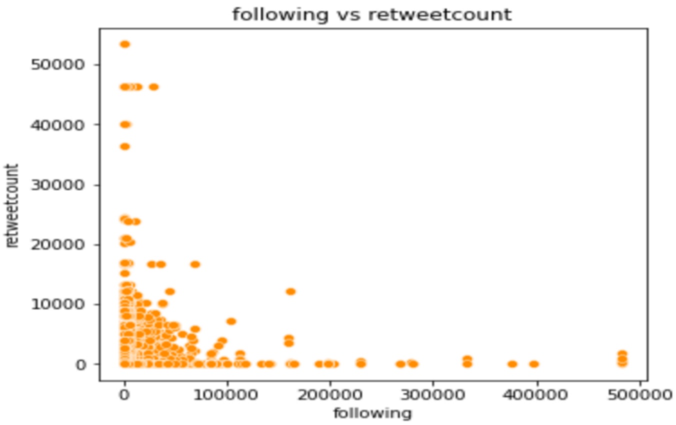


Fig. 2. Retweet Count versus total number of tweets

The two scatter plots shown in Figures 1 and 2 give us an idea of the popularity of a tweet based on the number of followers and total tweets posted by the user. We expect people with more followers and more activity (total tweets) to get more retweets. But the plots above show that distribution is very uneven and there is hardly any trend in both the cases. This calls for another feature to base our predictions on, which is the text of the tweets.

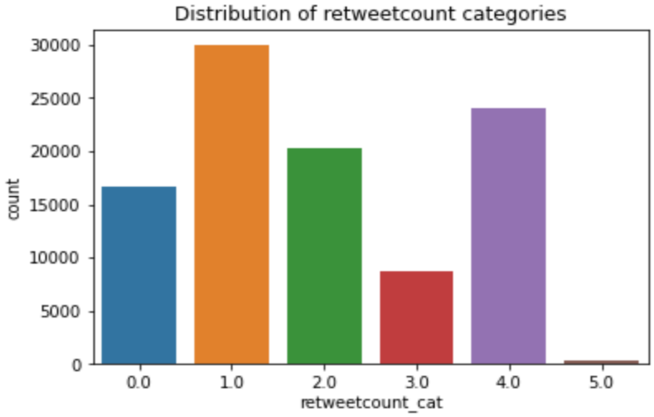


Fig.3: Distribution of retweetcount categories (labeled through bins)

The frequency of data in the different categories (bins) in Figure 3 show that it is imbalanced, though not very much. We could make it a balanced distribution using quantile cuts of the data. However, that would give us arbitrary values of the interval borders, not indicative of the popularity as we interpret in daily lives.

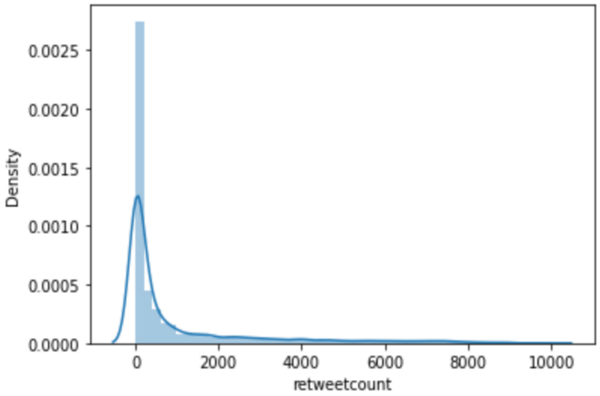
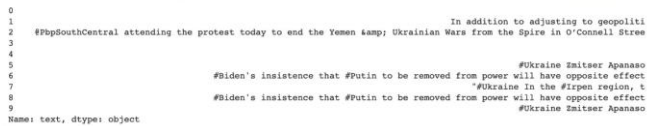


Fig.4: Distribution of retweetcount

Since we are also fitting regression models, it will be useful to visualize the distribution of the retweet count values. The above graph in Figure 4 shows the kernel density curve of the distribution. The plot has been made after filtering out outliers (retweet count > 10,000).

Figure 5 shows the first 10 unstructured text entries in the dataset. To process the text data, we needed to vectorize the texts for any mathematical model. Before doing so, we cleaned the text variable to remove html codes, punctuation marks, and English stop-words. We then performed stemming to reduce words to their root form (e.g., *invading* and *invasion* are converted to their root word *invade*). The data was then randomly split into training and testing sets in the ratio of 80:20, to test our models' accuracies.



```
0
1                                                                                    In addition to adjusting to geopoliti
2    #PbpSouthCentral attending the protest today to end the Yemen &amp; Ukrainian Wars from the Spire in O'Connell Stree
3
4
5                                                                                                 #Ukraine Zmitser Apanaso
6                                       #Biden's insistence that #Putin to be removed from power will have opposite effect
7                                                                                         "#Ukraine In the #Irpen region, t
8                                       #Biden's insistence that #Putin to be removed from power will have opposite effect
9                                                                                                 #Ukraine Zmitser Apanaso
Name: text, dtype: object
```

Fig.5: First 10 unstructured text entries in the data.

IV. Methods

We applied machine learning models on the dataset to predict our response variable, retweet count category (made based on the number of retweets). The predictors we used are *text* features (tweet text). To use the text variable as a predictor, we had to convert the text into vectors, using count vectorizer (bag-of-words) and TF-IDF methods.

Text Processing

- Bag-of-words: This is a method which converts the text data into numerical vectors which is what machine learning models can process. It makes every stemmed word after preprocessing a dimension in our vector space. The individual observations (tweets) are then given values in each 'dimension,' depending on how often a particular word exists. Naturally, since the dimension would be extremely high, and the tweets are 20-30 words long, we get sparse vectors in the process.

- TF-IDF: With bag-of-words, we do not have a method to provide weights to different words, which is critical to our analysis. This technique gives more weight to rare words and vice versa. It has a component that counts the relative frequency of a word in a record and logarithm of the inverse of the frequency in the entire dataset.

These vectors combined with the quantitative data have been used to train our models in both classification and regression settings. The following classification and regression models were used to predict the popularity of a tweet.

Classification and Regression

We created 5 bins with the 'retweetcount' column and performed classification to predict the target variable. Further, we performed regression to predict the 'retweetcount' column as it is, without creating any bins, thus making it a regression problem.

The evaluation metrics used for the models are given below:

Classification:

1. Accuracy – Accuracy is one of the metrics to evaluate classification models. It talks about what fraction of our data is correctly classified. Formally, accuracy is the ratio of number of correct predictions to total number of predictions. This metric is generally used when we have balanced data.

2. F1 Score - The F1-score evaluation metric for classification combines the precision and recall of a classifier by taking their harmonic mean. This metric is primarily used to compare the performance of two classifiers and is generally used when we have imbalanced data.

Regression:

1. RMSE - The Root Mean Squared Error measures the average magnitude of the errors. It talks about the deviation of predicted values from the actual values. The RMSE value with zero suggests that the model has a perfect fit, therefore, the lower the RMSE, the better the model and its predictions.

2. MAE - Mean absolute error (L1 loss) is one of the simplest loss functions and easily understandable regression evaluation metric. It is measured by averaging the absolute difference between the predicted values and the actual values across the data (arithmetic average of absolute errors). The lower the MAE, the higher the accuracy of a model.

- Logistic Regression: This method is a supervised classification algorithm which has a categorical dependent variable. While it acts like a classification problem in that it predicts one of two or more outcomes for the dependent variable, logistic regression is a regression model because it builds a regression model to predict the probability that a tweet belongs in the popular category. We extended the logistic regression model to classify retweet count into 5 bins. The testing accuracy of the logistic regression model in our analysis was 87.01%.

We started with logistic regression for its simplicity and to set a realistic benchmark for complex modeling techniques.

- Decision Trees: This method is a nonparametric, supervised algorithm that can be used for regression or classification. It can efficiently handle large, complicated data. It creates a model that predicts a target value by learning decision rules which depend on the predictors. This method gets its name because it classifies a population into "branches" that form an upside-down tree with a root node (population), internal nodes (features tests also known as branches), and leaf nodes (where final classes are assigned by majority vote). The testing 'accuracy' of the Decision Trees classification model for our data was 63.98%. And the testing 'rmse' of the Decision Trees regression model for our data was 1207.72.

- Random Forest Classification: This supervised algorithm that can be used for both regression and classification. Random forest, like the name suggests, consists of multiple individual decision trees that work together as an ensemble. Each decision tree in the random forest algorithm splits out a prediction and the class/category. For the classification problem, the class with the greatest number of votes becomes the final prediction for the model. Whereas, for the regression problem, the final prediction is the average of the prediction from each decision tree. The testing 'accuracy' of the Random Forest classification model for our data was 56.35%. And the testing 'rmse' of the Random Forest regression model for our data was 1188.67.

V. Results

The performance of each classification model is summarized in Table 1 below, and the performance of each regression model is summarized in Table 2 below.

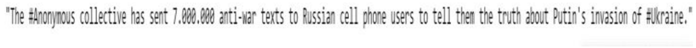
TABLE I. EVALUATION OF CLASSIFICATION MODELS

Model Name	**Training Accuracy (%)**	**Testing Accuracy (%)**	**Training F1 Score**	**Testing F1 Score**
Logistic Regression	89.57	87.01	89.13	86.68
Decision Trees	64.45	63.98	56.05	55.76
Random Forests	56.94	56.35	45.40	44.71

TABLE II. EVALUATION OF REGRESSION MODELS

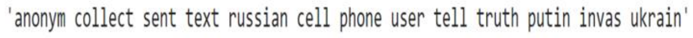
Model Name	**Training RMSE**	**Testing RMSE**	**Training MAE**	**Testing MAE**
Decision Trees	1148.57	1207.72	560.15	581.09
Random Forest	1240.60	1188.67	584.25	596.80

Our results indicate that for the classification modeling, the logistic regression model produced the highest testing accuracy score of 87.01%. This model performed significantly better than the decision trees and random forest. This indicates that the linear models could explain the variability in the data better and hence the tree-based models did not perform well and are not a good fit for our data (with the text data as the only predictor). A prominent reason for such a result could be the dimensionality of our data. Since the vectorized text features are high dimensional, models like random forest suffer from the curse of dimensionality. For our progress report, we considered fewer observations, which resulted in fewer dimensions in the vectorized texts and Decision Tree performed well. For the regression modeling, the Decision Trees performed the best with a testing rmse of 581.09, when compared to the performance of Random Forest. We have not performed Linear Regression here because we noticed that the distribution of response variable is not normally distributed. Looking at one observation, we start with our uncleaned text data, which looks like this:



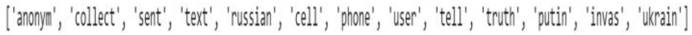
```
"The #Anonymous collective has sent 7.000.000 anti-war texts to Russian cell phone users to tell them the truth about Putin's invasion of #Ukraine."
```

After doing cleaning, stemming and stop-word removal, we get:



```
'anonym collect sent text russian cell phone user tell truth putin invas ukrain'
```

From the above data, we create tokens as under:



```
['anonym', 'collect', 'sent', 'text', 'russian', 'cell', 'phone', 'user', 'tell', 'truth', 'putin', 'invas', 'ukrain']
```

This is done on all the observations with each token acting as a dimension. The text data are then given values in each dimension depending on the frequency of eah token in their content. This creates a sparse vector, which is then transformed using IDF. Once we get the vectorized features, we fit the model on training set and make predictions using the model. We compare oour prediction with the original label and thus, calculate the accuracy.

VI. Conclusion and Future Work

Classification and regression models were analyzed to predict popularity of a tweet. Various models have been considered such as Logistic Regression, Decision Trees and Random Forest for classification, and Decision Trees, and Random Forest for regression. The classification model that performed the best in terms of having the highest testing accuracy was Logistic Regression. The regression model that performed the best in terms of having the lowest RMSE was Decision Trees. In this study, we learned how to process and analyze data in PySpark, which is an efficient tool to use when processing large data. We also learned how to process text data using natural language processing techniques. Because people can write whatever they want in their social media posts, we needed to find a way to structure the text data to conduct our analysis.

The model accuracies could further be improved by adding additional features such as number of followers an account user has, total tweets posted by an account user, date the tweet was posted, and hashtags contained in the tweet. A user with more followers could be likely to get more retweets, a user with more posted tweets could have a higher chance of having one of their tweets go viral, and the closer in time the tweet was posted to its related event could potentially impact prediction popularity. Hashtags are used to group content together regarding a specific topic. This makes it easier for people to find content they are interested in. Additionally, logarithmic transformation of the response variable could also be done to have the normal distribution. Furthermore, an emotion feature could be extracted from the text data, which can then be used for sentiment analysis and included in the predictive modeling to determine if popularity depends on the emotion of the message. Finally, our analysis can be refined for tweets that are in other languages besides English. Because we were using a Twitter dataset with posts about the Ukraine-Russia conflict, most of the tweets were in Ukrainian and Russian, and by including other languages in the analysis, there would be more data to include in the predictive models, which could improve accuracy.

References

[1] Rathord, Priyanka, Anurag Jain, and Chetan Agrawal. "A comprehensive review on online news popularity prediction using machine learning approach." trees 10.20 (2019): 50.

[2] Ren, He, and Quan Yang. "Predicting and evaluating the popularity of online news." Standford University Machine Learning Report (2015).

[3] Joshi, Mahesh, et al. "Movie reviews and revenues: An experiment in text regression." Human language technologies: The 2010 annual conference of the North American chapter of the Association for Computational Linguistics. 2010.

[4] Yu, Hai, Ying Hu, and Peng Shi. "A prediction method of peak time popularity based on twitter hashtags." IEEE Access 8 (2020): 61453-61461.

[5] Suh, Bongwon, et al. "Want to be retweeted? large scale analytics on factors impacting retweet in twitter network." 2010 IEEE second international conference on social computing. IEEE, 2010.

[6] Stepanova, Nataliya, et al. "Predicting Popularity of Polish Facebook Posts Using Author and Content Features."

[7] Wu, Bo, and Haiying Shen. "Analyzing and predicting news popularity on Twitter." International Journal of Information Management 35.6 (2015): 702-711.

[8] Daga, Ishita, et al. "Prediction of likes and retweets using text information retrieval." Procedia Computer Science 168 (2020): 123-128.

[9] Purtova, Daria. (February, 2022). Ukraine Conflict Twitter Dataset, Version 1. Retrieved April 6, 2022 from https://www.kaggle.com/datasets/bwandowando/ukraine-russian-crisis-twitter-dataset-1-2-m-rows/metadata.