Predicting Twitter Users' Political Orientation Final Project Report

CPSC 571 - Winter 2023
Group #7
Soumya Praveen Kumaria - 10124004
Se Yeon Sim - 10102631
Paruhang Basnet - 30021694
TA - Kashfia

Abstract

This research project aimed to apply machine learning training on three ML models (two base, one ensemble) on a dataset of over 85000 tweets from various users, with the aim of being able to accurately predict Twitter users political orientation (Republican or Democrat) based on their tweets. We chose Support Vector Machine (SVM) and Naive Bayes as our base models and an Ensemble classifier using SVM and Naive Bayes. For training and testing, we used an 80-20 split, where 80% of the dataset was used for training, and the remaining 20% was used for testing the efficacy of our trained models.The Ensemble classifier was able to match the accuracy of the base models predictions, the difference in accuracy between the base and ensemble model was not significant.

Introduction

Twitter, one of the largest social networking platforms, serves as a hub for users to share their thoughts on a wide range of topics. Major global events, such as elections, often prompt users to voice their opinions about candidates through tweets. Politicians also leverage Twitter for campaigning and increasing their public visibility *(Murthy, 2015).* Limited to 140 characters, tweets are concise and straightforward *(Lloret & Palomar, 2013).*

Text analysis of tweets can be employed to identify the intended message and ascertain whether the sentiment is positive or negative *(O'Connor et al., 2010).* By applying text classification using machine learning, we can analyze tweet content and determine which political candidate a user supports *(O'Connor et al., 2010).* Features such as hashtags, retweets, mentions etc. can be extracted from tweets and utilized for building machine learning models *(Van Vliet, Törnberg & Uitermark, 2020).* Through these models, election results can be predicted based on sentiment analysis, and popularity of a political candidate and the political alignment of Twitter users can be determined etc. *(Conover et al., 2011; Gaurav et al., 2013; Ali et al., 2022).*

For this research project, a dataset comprising 85,000 tweets from the 2018 U.S election is used to predict the political alignment of Twitter users using machine learning models. Support Vector Machine (SVM) and Naïve Bayes Classifiers are employed to train and test the dataset. Subsequently, the two classifiers are combined to create an ensemble classification model for predicting users' political alignment. Finally, the three classifiers are compared to identify the most effective model for predicting the political alignment of Twitter users.

Overview

Social media platforms, particularly Twitter, have become increasingly important in political campaigns and discours. Support Vector Machines (SVM) and Naive Bayes models were analyzed, in their effectiveness for predicting user political alignment and sentiment analysis of tweets. The research on SVM showed a 91% accuracy in political alignment prediction and could be improved to 95% when incorporating retweet networks, while Naive Bayes reached

75% accuracy in sentiment analysis and enabled accurate prediction of election outcomes at the state level *(Conover et al., 2011)*.

Support Vector Machines (SVM):

Using the Twitter “gardenhose” streaming API, researchers collected a sample of politically relevant tweets based on co-occurrence discovery procedures with seed hashtags. A dataset of 1000 randomly selected users were annotated as 'left', 'right', or 'ambiguous' by two independent annotators, resulting in 373 left, 506 right, and 77 ambiguous users *(Conover et al., 2011).* Applying SVM with TFIDF on full text and partial hashtags, they achieved 91% overall accuracy in predicting user political alignment as 'left' or 'right'. Incorporating retweet networks further improved classification accuracy to 95% *(Conover et al., 2011).* The study also identified the most tweeted websites and hashtags by political alignment, noting that politically active Twitter users do not always visit sites with higher overall traffic, instead sometimes choosing to browse select sites aligning more with their political sentiments *(Conover et al., 2011)*.

Naive Bayes:

Researchers collected tweets from politically ambiguous states (Florida, Ohio, North Carolina), then using known polarizing hashtags such as “#imwithher” and “#maga” as seeds, they built a Naïve Bayes machine learning model to try and accurately predict election outcomes in the chosen states *(Oikonomou & Tjortjis, 2018)*.

Using Naive Bayes model with a known accuracy of 75%, sentiment analysis was performed to classify tweets as positive, negative, or neutral *(Oikonomou & Tjortjis, 2018)*. The model was trained on 1.5 million pre-classified tweets. The Python library “Textblob” was implemented to add another layer of text analysis to help increase accuracy. Then, sentiment scores were calculated based on the frequency of positive, negative, and neutral words in each tweet.

From the general sentiment towards each candidate in the selected states, predictions were made by the model for the winner of the election at the state level. Despite varying probability percentages, all the election winners were accurately predicted by the model *(Oikonomou & Tjortjis, 2018)*.

The papers demonstrate the effectiveness of SVM and Naive Bayes models in analyzing political sentiments on Twitter. SVM proved highly accurate in predicting user political alignment, while Naive Bayes was effective in sentiment analysis and election outcome prediction *(Conover et al., 2011; Oikonomou & Tjortjis, 2018).*

Discussion of Work Done

The corpus on using machine learning models to analyze twitter data is large and much of the common models have already been researched and tested.

This study involves the use of SVM and Naive Bayes as our base models to create an ensemble model using voting technique. Furthermore, performance of all the models is compared and analyzed.

Attempts at improving the accuracy of the supervised models largely involved implementing additional features into the base models. This study involves combining base models into an ensemble model that outperforms its base models. The ensemble model usually has higher performance than individual models as they combine the predictive power of multiple base models.

Dataset

The dataset chosen for this project is the "Democrat vs Republican Tweets" dataset, available on Kaggle. This dataset was selected primarily because of its relevance, as it contains a large number of tweets from Democratic and Republican politicians, making it highly suitable for predicting political orientation based on tweet content. The dataset is almost evenly distributed as you can see in Figure 1, with Democrats tweets accounting for 48.7% and Republicans tweets making up 51.3%.

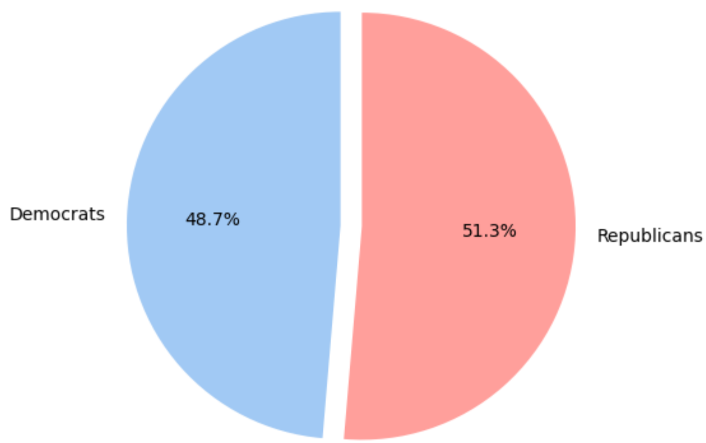

Figure 1. Distribution of Tweets by Party

Additionally, the dataset comes with pre-assigned labels for political orientation (Democratic or Republican), allowing us to focus on model development and evaluation without having to manually label the data. A representation of the dataset can be seen in Table 1. The dataset is also reasonably current, containing tweets posted up until September 2021, and offers a sufficiently large sample size of over 85,000 tweets for training and evaluating machine learning models. Moreover, the dataset includes tweets from various politicians, capturing a diverse range of political viewpoints and language patterns, which can help our models generalize better.

Table 1. Representation of the raw dataset

Party	Handle	Tweet
Democrat	RepDarrenSoto	RT @NALCABPolicy: Meeting with @RepDarrenSoto ...
Democrat	RepDarrenSoto	RT @Vegalteno: Hurricane season starts on June...
Republican	RepTomPrice	Check out my op-ed on need for End Executive O...
Republican	RepTomPrice	Yesterday, Betty & I had a great time lear...

However, there are some limitations to the dataset. For instance, it focuses on tweet content and does not provide additional features related to the users, such as their location, follower count, or account age. Including such features might have improved the predictive performance of the models. Lastly, as with any dataset containing real-world data, the tweets may contain noise such as retweets, URLs, and special characters, which can affect the quality of the data and require additional preprocessing steps. Figure 2 and 3 show the word cloud for Democrats and Republican tweets respectively.

Figure 2. Word Cloud of Republican Tweets

Figure 3. Word Cloud of Democrat Tweets

Preprocessing

Preprocessing text data is an essential step in optimizing our dataset for machine learning algorithms. To achieve this, a series of techniques to transform the raw text data into a structured and informative format were employed which can be seen in Figure 4.

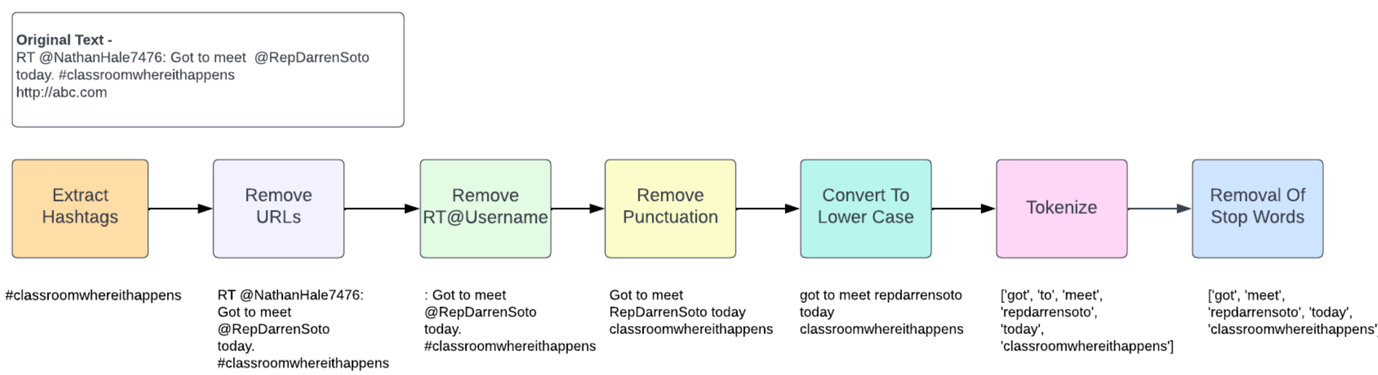

Figure 4. Tweets' pre-processing Steps

The first step is to extract hashtags from the text by identifying words that begin with the "#" symbol. Hashtags provide valuable insights into the topics and themes of tweets, making it crucial to extract them for analysis. URLs and retweet information from the text data is removed next. This step ensures that the focus remains on the main content of the tweet and removes any irrelevant or redundant information. Next, unnecessary punctuation marks are removed from the text to standardize the format of the data. This simplifies the analysis and reduces noise in the dataset.Afterward, the text is tokenized, breaking it down into individual words, and converting the individual words to lowercase to standardize the format. This step ensures that words with different capitalizations are treated the same, ensuring consistency in the dataset. Finally, stop words and specific words are removed from the tokens, reducing noise and focusing on relevant information for our classification task. Once all the preprocessing steps are complete, filtered tokens and hashtags are rejoined back into strings, separating words with a space.

These steps ensure that the text data is in a usable format for our machine learning algorithms, enabling them to learn patterns and relationships between features and class labels accurately. By employing these techniques, the dataset can be optimized for machine learning and accuracy of the predictions can be improved.

Vectorization

TfidfVectorizer stands for Term Frequency-Inverse Document Frequency Vectorizer. It is a technique used in natural language processing to convert text data into a numerical format suitable for machine learning algorithms. TfidfVectorizer assigns a weight to each word in the document based on its term frequency (TF) and inverse document frequency (IDF). The term frequency represents the number of times a word appears in a document, while the inverse document frequency represents the importance of a word across multiple documents *(GeeksforGeeks, 2023)*.

Machine learning algorithms, such as Naive Bayes and Support Vector Machines (SVM), require numerical input data to function. Since the text data is not numerical, it needs to be transformed into a numerical format that can be used as input for these algorithms *(Bedi, 2018)*. TfidfVectorizer helps in achieving this by converting the raw text data into a matrix of TF-IDF features. The resulting matrix represents the text data in a way that emphasizes the importance of specific words within the context of the entire document collection *(Bedi, 2018).* This allows machine learning algorithms, like Naive Bayes and SVM, to make better predictions by taking into account the relative importance of words in the dataset.

SVM

Support Vector Machines (SVM) is a supervised machine learning algorithm used for classification and regression tasks. It works by finding the optimal hyperplane that best separates the data points into different classes *(Ray, 2023)*. SVM can handle both linear and non-linear problems by using different kernel functions, such as linear, polynomial, or radial basis function (RBF) kernels. SVM effective in high-dimensional data *(Ray, 2023)*.

LinearSVC, or Linear Support Vector Classification, is a variant of the SVM algorithm that employs a linear kernel. It is specifically designed for binary and multi-class classification problems where the decision boundary between classes is linear *(Pedregosa et al., 2011).* LinearSVC is also well-suited for high-dimensional data features and provides a computationally efficient alternative to SVM with linear kernels *(Pedregosa et al., 2011)*.

LinearSVC was selected as the preferred method for predicting the political orientation of Twitter users, owing to its ability to handle large feature spaces, offer scalability and efficiency. Predicting the political orientation from tweets involves managing high-dimensional feature spaces with diverse words and phrases, and LinearSVC is proficient in processing such data without compromising its performance *(Pedregosa et al., 2011).*

Moreover, LinearSVC's algorithm, implemented by 'liblinear', outperforms its 'libsvm-based' SVC counterpart in terms of efficiency, particularly when working with linearly separable data. It can handle millions of instances and scale almost linearly to millions of samples and/or features *(Pedregosa et al., 2011)*, making it an ideal candidate for large-scale social media analysis.

Naive Bayes

Naive Bayes is a probabilistic machine learning algorithm that leverages the Bayes theorem for classification tasks. It is deemed "naive" because it assumes that the features used for classification are conditionally independent given the class label *(Pedregosa et al., 2011)*.

Multinomial Naive Bayes is a variant of the Naive Bayes algorithm, specifically designed for discrete data such as word counts or term frequencies in text classification problems *(Pedregosa et al., 2011)*. It calculates the conditional probability of each feature given the class label based on the frequency of the feature's occurrence within each class *(Pedregosa et al., 2011).*

Multinomial Naive Bayes was utilized for predicting the political orientation of Twitter users because of its effectiveness in handling text data. Text classification tasks like predicting political orientation from tweets require handling discrete data such as word counts or term frequencies. Multinomial Naive Bayes is suitable for such tasks and provides relatively high accuracy predictions by capturing the relationship between words and class labels *(Pedregosa et al., 2011).*

Ensemble Model

An ensemble model is a machine learning approach that utilizes multiple models' predictions to generate a more accurate final prediction *(Brownlee, 2021).* For combining multiple models' predictions, voting ensemble technique was utilized. Voting ensemble technique involves taking a majority vote. There are two ways to perform this technique: hard voting and soft voting *(Brownlee, 2021).*

The soft voting technique was utilized to develop an ensemble model, which involves averaging the predicted probabilities of each model and selecting the class label with the highest average probability as the final prediction *(Kumar, 2020).* For example, if multiple models were trained to predict a Twitter user's political orientation based on their tweets, the soft voting approach would compute the average probability for each class and select the one with the highest average probability.

During the implementation phase of the ensemble model, SVMs present a challenge as they do not provide well-calibrated probability by default. To address this issue, the CalibratedClassifierCV function was employed. This function calibrates the predicted probabilities of an SVM classifier to ensure their accuracy and consistency with other classifiers in the ensemble model *(Pedregosa et al., 2011).*

Contribution

SVM and Naïve Bayes are the two most used models to predict a Twitter user's political orientation and for sentimental analysis of tweets as they typically deliver high levels of accuracy (*Conover et al., 2011; Singh et al., 2021)*. Theoretically, combining these two models should produce even better results in predicting a Twitter user's political orientation.

In previous research, ensemble classifiers have been created by combining various classifiers for sentiment analysis (*Saleena, 2018; Jose & Chooralil, 2016; Chang & Wang, 2020)*. However, ensemble classifiers have not been extensively explored for predicting a Twitter user's political orientation. Through this study, we aim to investigate whether, in practice, ensemble classifiers can outperform individual classifiers in accurately predicting a Twitter user's political orientation.

Results

As the testing set consisted of 20% of the dataset, around 17,000 tweets were tested to determine the performance of each model. All the models were trained and tested on the same training and testing set. Confusion matrix for each model was created.

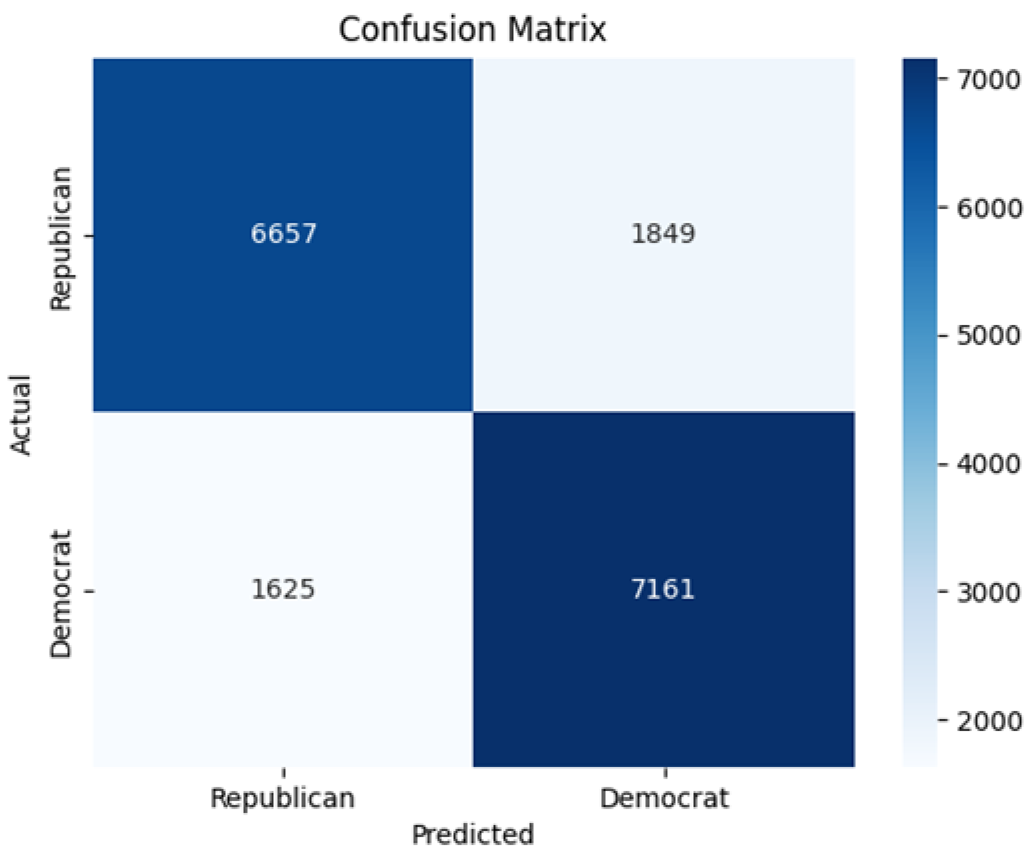

Figure 5. Confusion Matrix for Support Vector Machine (SVM)

In Figure 5, there are 6,657 tweets that are true positive this means that 6657 tweets were predicted as Republicans and they were actually republicans. 7,161 tweets are true negative that means 7161 tweets were classified as Democrat tweets and they were actually Democrats tweets.

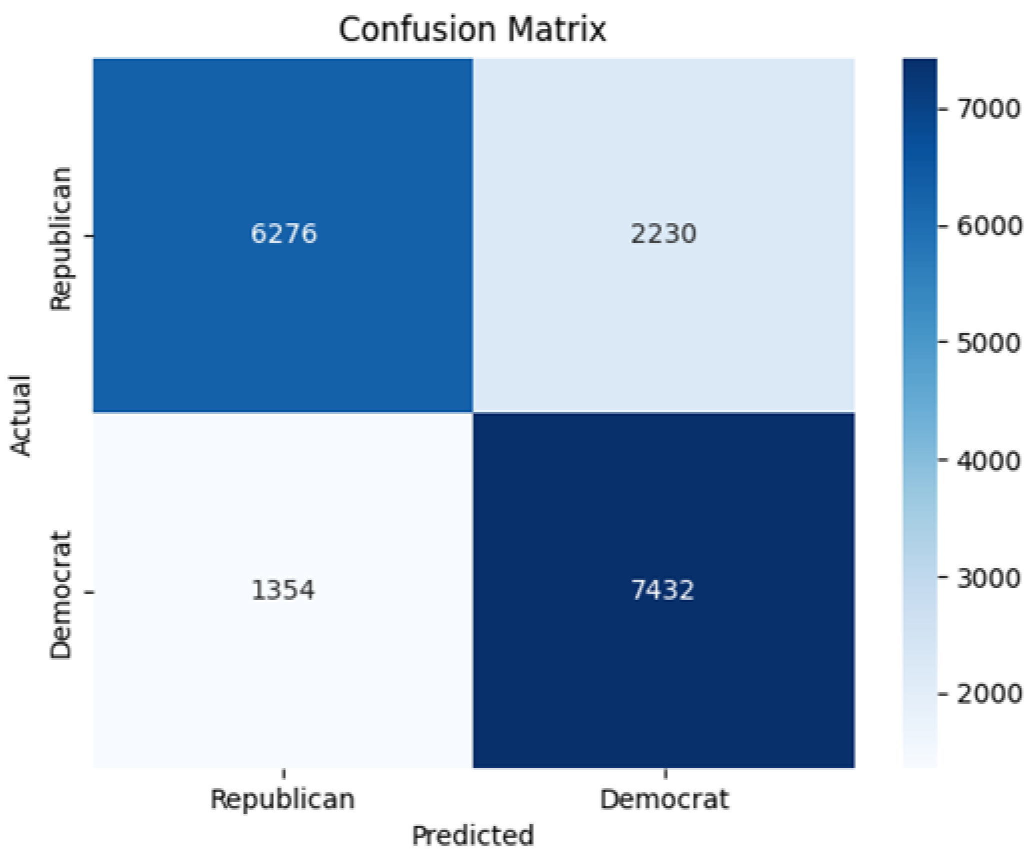

Figure 6. Confusion Matrix for Naive Bayes

In Figure 6, there are 6,276 tweets that are true positive this means that 6,276 tweets were classified as Republican tweets and they were actually Republican tweets. 7,432 tweets are true negative that means 7,432 tweets were classified as Democrat tweets and they were actually Democrat tweets.

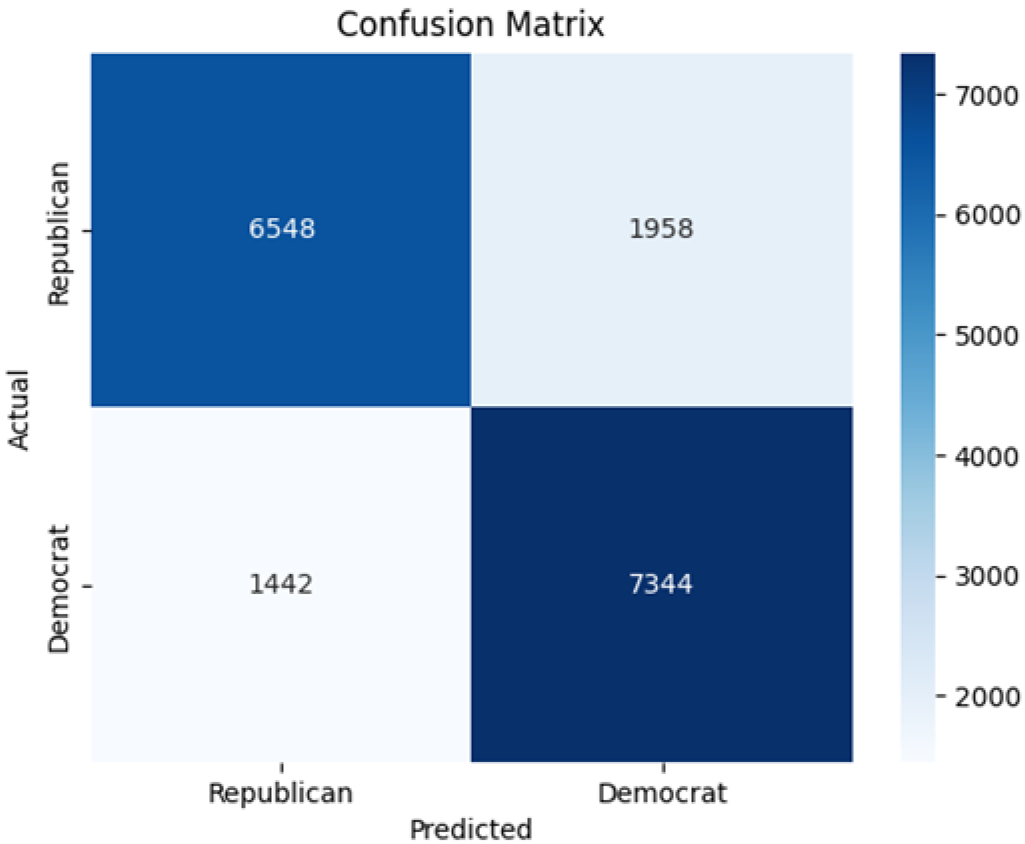

Figure 7. Confusion Matrix for Ensemble

In Figure 7, there are 6,548 tweets that are true positive this means that 6,548 tweets were classified as Republican tweets and they were actually Republican tweets. 7,344 tweets are true negative, 7,344 tweets were classified as Democrat tweets and they are actually Democrat tweets.

F1- scores, precision, recall and accuracy are the metrics used to compare performance of each classifier. These results obtained by all the classifiers provide insights into the performance of each model in predicting the political orientation of Twitter users based on their tweets.

Table 2. Comparison of model performance

Model	Party	Precision	Recall	F1-Score	Accuracy
SVM	Democrat	80.38%	78.26%	79.31%	
	Republican	79.48%	81.50%	80.48%	79.99%
Naïve Bayes	Democrat	82.25%	73.78%	77.79%	
	Republican	76.92%	84.59%	80.57%	79.27%
Ensemble	Democrat	81.95%	76.98%	79.39%	
	Republican	78.95%	83.59%	81.20%	79.91%

Precision measures the proportion of true positives among the predicted positives, indicating the model's ability to correctly identify positive instances *(Powers, 2020)*. A higher precision implies that the model is more effective at minimizing false positives. In this case, Naïve Bayes has a precision of 82.25% for Democrats which is higher than SVM and Ensemble. Naïve Bayes and is better at correctly predicting positive instances for Democrats. Similarly, for Republicans SVM is better at predicting positive instances as they have a precision of 79.48%.

Recall, on the other hand, measures the proportion of true positives among the actual positives, reflecting the model's ability to identify all the positive instances in the dataset *(Powers, 2020)*. In this case, SVM has the highest recall value of 78.26% for Democrats. This means that SVM has the highest proportion of true positives among actual positives for Democrat tweets. Similarly Naïve Bayes with 84.59% has the highest proportion of true positives among actual positives for Republican tweets.

F1-score is a metric that combines both precision and recall, providing a balanced measure of the model's performance *(Powers, 2020)*. A higher F1-score indicates that the model is more effective at achieving a balance between precision and recall. In this case, Ensemble has the highest F1 score of 79.39% for Democrats suggesting that Ensemble performs better as compared to Naïve Bayes and SVM for predicting Democrat tweets. Similarly, Ensemble performs better as compared to SVM and Naive Bayes for predicting Republican tweets as it has the highest F1-score of 81.20%.

The accuracy represents the proportion of correctly classified instances out of the total instances in the dataset *(Powers, 2020)*. A higher accuracy implies that the model is better at making correct predictions. The accuracy for both SVM and ensemble is almost 80% and the accuracy for Naïve Bayes is 79.27% indicating that SVM and Ensemble are slightly better at predicting political orientation. SVM is slightly better than Ensemble as it's accuracy is 79.99% whereas Ensemble classifier has an accuracy of 79.91%.

Possible Reasons for Limited Improvement Observed in our Ensemble Model

The limited improvement observed in the ensemble model can be attributed to three possible reasons. One possible explanation for the limited improvement is the similar performance of the Naive Bayes and SVM models. Both classifiers achieved approximately 80% accuracy on their own, which might have led to only marginal gains when combined in an ensemble.

Secondly, both the Naive Bayes and SVM models made similar errors when classifying tweets. Approximately 70% of the tweets misclassified by the Naive Bayes model were also misclassified by the SVM model, indicating an overlap in misclassifications that limited the potential improvement an ensemble model could provide.

Thirdly, the dataset used contained retweets which could have led to the similar performance of all the models as each retweet would be classified the same. There might be some duplicated tweets in the form of retweets in the dataset that could cause performance of the models to be the same,

Finally, the lack of diversity in the base classifiers may have contributed to the limited improvement. Ensemble models typically benefit from combining predictions of diverse models, which can compensate for each other's weaknesses. However, if the base classifiers share similar underlying assumptions, errors, and biases, the ensemble model might not be able to leverage their complementary strengths, leading to limited improvement in performance *(Nam et al., 2021).*

Summary

Twitter is a popular social media site. Twitter users use the site to express their opinions through tweets. Twitter is vastly used during elections. This study compared performance of three machine learning models that can be used to predict a twitter user's political orientation. Support Vector Machine (SVM), Naive Bayes and Ensemble model that used voting technique are the three models compared in this study. SVM and Naive Bayes acted as base classifiers for the ensemble model. Voting ensemble models are expected to make more accurate predictions as they combine predictions of multiple base models *(Leon et al., 2017).* Soft voting is used by the ensemble model to predict class labels as it classifies labels based on predicted probabilities *(Kumari et al, 2021)*. The voting ensemble model didn't outperform its base models even though it combined predictions from its base model. All the models have similar performance. SVM, Naive Bayes and Ensemble models have accuracy of 79.99%, 79.27% and 79.91% respectively. This behavior can be attributed to similar performance of the base classifiers and the lack of diversity of base classifiers. Around 70% of the tweets that were misclassified by

Naive Bayes model were also misclassified by SVM. Only two base classifiers are used in this study and both of them are supervised machine learning classifiers that could have also limited the performance of the ensemble model.

Conclusion

The purpose of this research was to identify an effective machine learning that could predict political leaning of a twitter user through their tweets. Support Vector Machine (SVM), Naive Bayes and Ensemble Voting models were used in this study. It can be concluded that all three models have the same performance because SVM, Naive Bayes and Ensemble models have accuracy of 79.99%, 79.27% and 79.91% respectively. Both the Naive Bayes and SVM models made similar errors when classifying tweets. As the Ensemble model was created by combining the predictions of both SVM and Naive Bayes, it made the same errors as them and no significant improvement in the performance was observed.

Future Work

Future research work can be done on dataset size and adding more machine learning models. The dataset used in this study contained tweets above 85,000. A larger dataset might be beneficial as it could provide better insights into each model's performance.

SVM and Naive Baye are the two supervised machine learning models used in this study. Usage of some unsupervised machine learning models could help improve the performance of ensemble model. Using multiple machine models would further help to improve the performance of ensemble classifiers. Different ensemble machine learning models should be compared as well. In this study, only ensemble voting classifiers is used. Boosting, bagging and stacking are other ensemble models that can be used.

References

Nam, G., Yoon, J., Lee, Y., & Lee, J. (2021). Diversity matters when learning from ensembles. *Advances in neural information processing systems*, *34*, 8367-8377.

Powers, D. M. (2020). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*

Leon, F., Floria, S. A., & Bădică, C. (2017). Evaluating the effect of voting methods on ensemble-based classification. In *2017 IEEE international conference on INnovations in intelligent Systems and applications (INISTA)* (pp. 1-6). IEEE.

Kumari, S., Kumar, D., & Mittal, M. (2021). An ensemble approach for classification and prediction of diabetes mellitus using soft voting classifier. *International Journal of Cognitive Computing in Engineering*, *2*, 40-46.

GeeksforGeeks. (2023). *Understanding TF-IDF (term frequency-inverse document frequency)*. GeeksforGeeks. Retrieved April 12, 2023, from https://www.geeksforgeeks.org/understanding-tf-idf-term-frequency-inverse-document-frequency/

Bedi, G. (2018). *Simple guide to text classification(nlp) using SVM and Naive Bayes with python*. Medium. Retrieved April 12, 2023, from https://medium.com/@bedigunjit/simple-guide-to-text-classification-nlp-using-svm-and-naive-bayes-with-python-421db3a72d34

Ray, S. (2023). *Learn how to use support vector machines (SVM) for Data Science*. Analytics Vidhya. Retrieved April 12, 2023, from https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/

Brownlee, J. (2021). *How to develop voting ensembles with python*. MachineLearningMastery.com. Retrieved April 12, 2023, from https://machinelearningmastery.com/voting-ensembles-with-python/

Kumar, A. (2020). *Hard vs soft voting classifier python example*. Data Analytics. Retrieved April 12, 2023, from https://vitalflux.com/hard-vs-soft-voting-classifier-python-example/

Conover, M. D., Gonçalves, B., Ratkiewicz, J., Flammini, A., & Menczer, F. (2011). Predicting the political alignment of twitter users. In *2011 IEEE third international conference on privacy, security, risk and trust and 2011 IEEE third international conference on social computing* (pp. 192-199). IEEE.

Oikonomou, L., & Tjortjis, C. (2018, September). A Method for Predicting the Winner of the USA Presidential Elections using Data extracted from Twitter. In *2018 South-Eastern European*

Design Automation, Computer Engineering, Computer Networks and Society Media Conference (SEEDA_CECNSM) (pp. 1-8). IEEE.

O'Connor, B., Balasubramanyan, R., Routledge, B., & Smith, N. (2010). From tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of the international AAAI conference on web and social media* (Vol. 4, No. 1, pp. 122-129).

Murthy, D. (2015). Twitter and elections: are tweets, predictive, reactive, or a form of buzz?. *Information, Communication & Society*, 18(7), 816-831. DOI: 10.1080/1369118X.2015.1006659

Lloret, E., & Palomar, M. (2013). Towards automatic tweet generation: A comparative study from the text summarization perspective in the journalism genre. *Expert Systems with Applications*, 40(16), 6624-6630.

Gaurav, M., Srivastava, A., Kumar, A., & Miller, S. (2013). Leveraging candidate popularity on Twitter to predict election outcome. In *Proceedings of the 7th workshop on social network mining and analysis* (pp. 1-8).

Van Vliet, L., Törnberg, P., & Uitermark, J. (2020). The Twitter parliamentarian database: Analyzing Twitter politics across 26 countries. *PLoS one*, 15(9), e0237073.

Ali, H., Farman, H., Yar, H., Khan, Z., Habib, S., & Ammar, A. (2022). Deep learning-based election results prediction using Twitter activity. *Soft Computing*, 26(16), 7535-7543

Pedregosa, F., Varoquaux, Ga"el, Gramfort, A., Michel, V., Thirion, B., Grisel, O., … others. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*(Oct), 2825–2830.

Saleena, N. (2018). An ensemble classification system for twitter sentiment analysis. *Procedia computer science*, *132*, 937-946.

Jose, R., & Chooralil, V. S. (2016). Prediction of election result by enhanced sentiment analysis on twitter data using classifier ensemble Approach. In *2016 international conference on data mining and advanced computing (SAPIENCE)* (pp. 64-67). IEEE

Chang, C., & Wang, X. (2020). Research on Dynamic Political Sentiment Polarity Analysis of Specific Group Twitter Based on Deep Learning Method. In *Journal of Physics: Conference Series* (Vol. 1651, No. 1, p. 012108). IOP Publishing.

Singh, A., Dua, N., Mishra, V. K., Singh, D., & Agrawal, A. (2021). Predicting Elections Results using Social Media Activity A Case Study: USA Presidential Election 2020. In *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)* (Vol. 1, pp. 314-319). IEEE.