Detecting Fake News: Binary Classification of Fake vs. True News based on Textual Characteristics of News Articles

Sarah Gauthier, Jessie Liang and Vinay Valson

2025-11-19

Table of contents

1 Summary

News is everywhere. In this digital era, there exists both true and fake news on the Internet, making it harder for readers and news agents to differentiate them. This makes fake news detection a technology with rising demands to help defend information integrity. To solve this problem, applied machine learning is used for binary classification. The fitted Naive Bayes model with `MultinomialNB` as estimator and `title`, `text` and `subject` as features yields a test classification accuracy of 0.954, where the false positive and false negative rates are both low. The AP score and AUC value of the final model end up being 0.97 and 0.98 respectively, suggesting decent and satisfactory performance. Compared to the baseline dummy classifier, this Naive Bayes model classifies both true and fake news significantly better, which is reflected by consistently high and similar training, cross-validation and test scores.

This machine learning model shows good performance, which highlights its potential to be deployed and used in the real world. Despite this, it does have limitations since the training data does not cover all possible types of news. When deployed in the wild, the model might perform badly in news of other realms, such as entertainment news. If more comprehensive data can be collected, then this model might handle various cases more accurately.

2 Introduction

"Fake news", also known as disinformation, is deliberately misleading or false information presented as legitimate news (Canada 2024). Its rapid spread in recent years has become a pressing concern, particularly with the rise of social media platforms that accelerate the dissemination of content. These platforms are designed with the primary objective of capturing and retaining user attention for as long as possible (Pennycook and Rand 2021). To that end, fake news often exploits this design to gain traction.

There is high public awareness of this issue in Canada. In 2023, Statistics Canada reported that 59% of Canadians admitted that they were "very or extremely concerned about online misinformation" in the Survey Series on People and their Communities (Bilodeau and Khalid 2024). In the United States, the concept of "fake news" grew in popularity during the 2016 presidential election. Allcott and Gentzkow (2017) estimated that false news stories were shared on Facebook at least 38 million times in the 3 months leading up to the election. Such widespread circulation of this false information can have serious consequences, including

influencing public opinion and undermining trust in legitimate news sources. Thus, methods for detecting fake news may be beneficial in mitigating its spread.

This report investigates whether a machine learning model can accurately classify news articles as "fake" or "true" based on their textual characteristics, such as the article's title, subject and body content. Automated detection of fake news has the potential to mitigate the spread and impact of information, particularly in social media's fast-paced environment where information is shared instantly and at scale, from numerous sources.

3 Data

The dataset used in this analysis is the "Fake and Real News Dataset" available on Kaggle (Bisaillon 2023). It contains approximately 40,000 news articles published in the United States between 2015 and 2018. Each row in the dataset represents a news article, with a column for its title, subject, date, and body text.

3.1 Data Cleaning and Transforming

The raw data from Kaggle was initially provided as two seperate datasets: one for the fake articles and one for the true articles. To prepare the data for modeling, we conducted some validation checks to make sure the data was in the expected format, with no missing observations or duplicate entries. We then cleaned and standardized the data before combining the true and fake datasets into one complete dataset. A new target column was added to label each article as either True or Fake, and the subject column values were simplified into consistent categories (political vs. non-political) so that the merged dataset has consistent features ready for analysis.

To support model training and evaluation, we shuffled the rows and split the data into training (80%) and testing (20%) sets. Further validation was conducted on the training set to identify any outliers and anomoulous correlations. Finally, we conducted exploratory data analysis (EDA) on the training data set to examine the distribution of the target variable and gain initial insights into the dataset.

4 Methods

4.1 Exploratory Data Analysis (EDA)

To begin our exploratory analysis of the training data, we generated a concise summary of the data, to show the non-null value counts and data types for each feature in the training dataset. This was done using the Pandas `.info()` method.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30903 entries, 0 to 30902
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   title    30903 non-null  object
 1   text     30903 non-null  object
 2   subject  30903 non-null  object
 3   date     30903 non-null  object
 4   target   30903 non-null  object
dtypes: object(5)
memory usage: 1.2+ MB
None
```

From this summary, we see that the training data contains 30903 observations. Each observation represents a news article with a title, text, subject, date and target label. Notably, we also see that there are no missing values in our training data set which means we can proceed without any imputation or row removal.

4.1.1 Count of Fake vs. Real News Articles

To better understand the balance of the target classes, we plotted the counts of articles labeled as `True` and `Fake` in the training dataset.

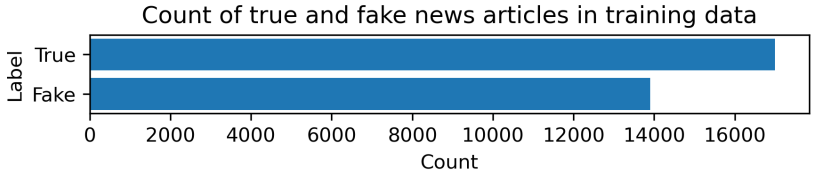

Figure 1: Count of true vs. fake news articles in training dataset

From Figure 1, we can see that the training dataset is fairly balanced, with 13905 fake news articles and 16998 real news articles. This balance may help to prevent bias towards one class over the other when training our model and increases the likelihood that it will learn to distinguish between fake and real news effectively.

4.1.2 Word Clouds for Title Column

Beyond class counts, it is useful to explore the language patterns in the dataset. Word clouds provide a quick visual summary of the most frequent words appearing in article titles. By com-

paring words appearing in fake and real news titles, we can identify differences in vocabulary, tone, or emphasis that may help models learn discriminative features.

Before generating the word clouds, we cleaned the text by removing punctuation and possessive contractions (e.g., 's). This ensures that the visualization highlights more meaningful words.

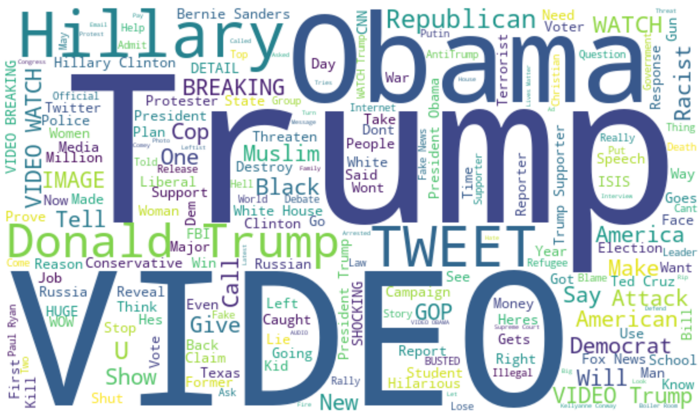

Figure 2: Word cloud for titles of fake news articles

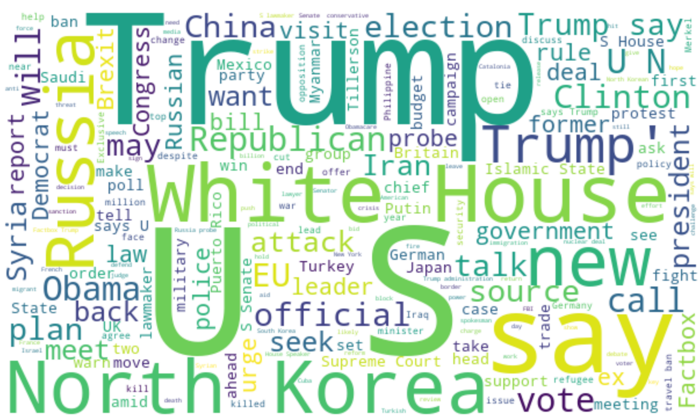

Figure 3: Word cloud for titles of true news articles

The most prominent words in the fake news titles include "VIDEO", "Trump" and "Obama". The most common words in the true news titles include "Trump", "US" and "White House".

We can see that "Trump" appears frequently in both fake and true news titles, which means that it may not give much discriminatory power when classifying articles. However, words like "VIDEO", that appear much more frequently in fake news titles, may help the model identify fake news articles.

4.1.3 Word Clouds for Text Column

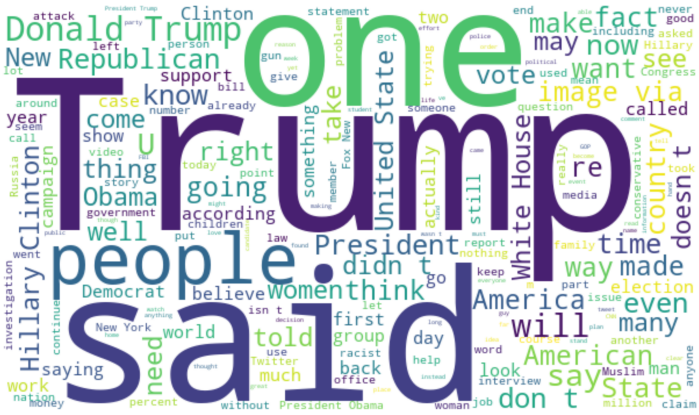

Figure 4: Word cloud for text of fake news articles

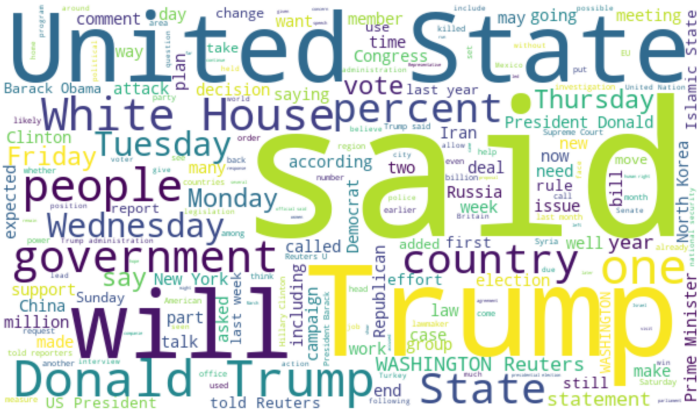

Figure 5: Word cloud for text of true news articles

The most commonly used words in the body text of fake news articles include “Trump”, “said”, and “one”. The most frequent words in true news articles are “Trump”, “said”, and “United State”. Given that there is more overlap in the common words used in the body text of fake and true news articles compared to the titles, the titles may provide more distinctive features for classification.

4.1.4 Comparing Title and Text Length between Fake and Real News Articles

Another important aspect of exploratory analysis is to examine whether simple structural features of the articles, such as the length of titles and body text, differ between fake and real

news. To do this, we generated histograms that show the distribution of article title lengths and article text lengths for fake and true news.

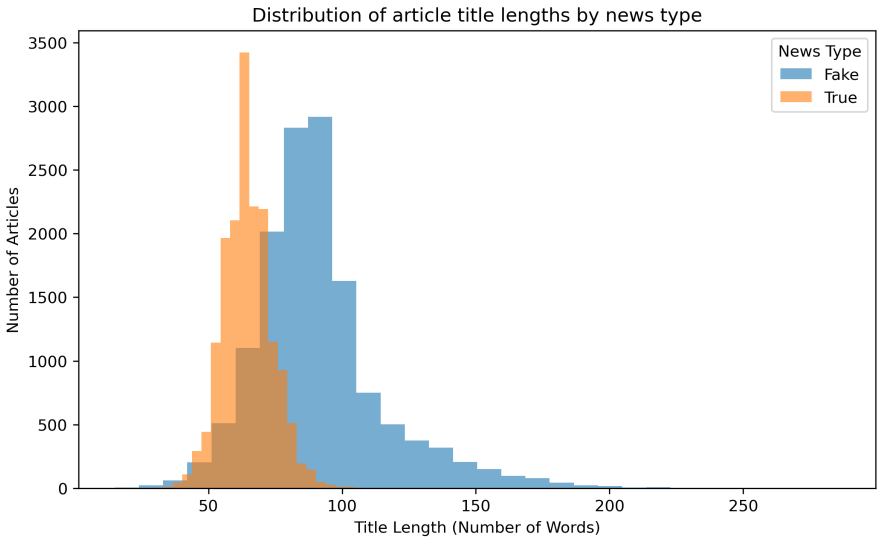

Figure 6: Distribution of article title lengths for fake and true news articles

From Figure 6, we can see that the length of true news titles appears to have bell-shaped distribution, while the length of fake news titles is right-skewed. This suggests that if the title of a news article is particularly long, it may be more likely to be fake news. This strategy may be utilized to grab readers' attention with sensational or clickbait-style headlines.

When comparing the body text lengths in Figure 7, we can see that both fake and true news articles have right-skewed distributions. Their distributions are quite similar, indicating that body text length may not be a strong indicator of whether an article is fake or true news.

4.1.5 Comparing the Percentage of Counts of Subjects between Fake and Real News Articles

Another feature of our dataset is the subject of the news article. Here, we examine whether the suject matter of articles differs between fake and real news. In our dataset, each article is labeled as either *political* or *non-political*. To compare distributions fairly, we calculated the percentage of articles in each subject category within the fake and true subsets of the training data and plotted the results.

As shown in Figure 8, non-political news accounts for a larger portion of the fake news, while there is a larger percentage of political news in the true news. The difference of percentages exists but is not conspicuous, suggesting this engineered feature `subject` may still be useful

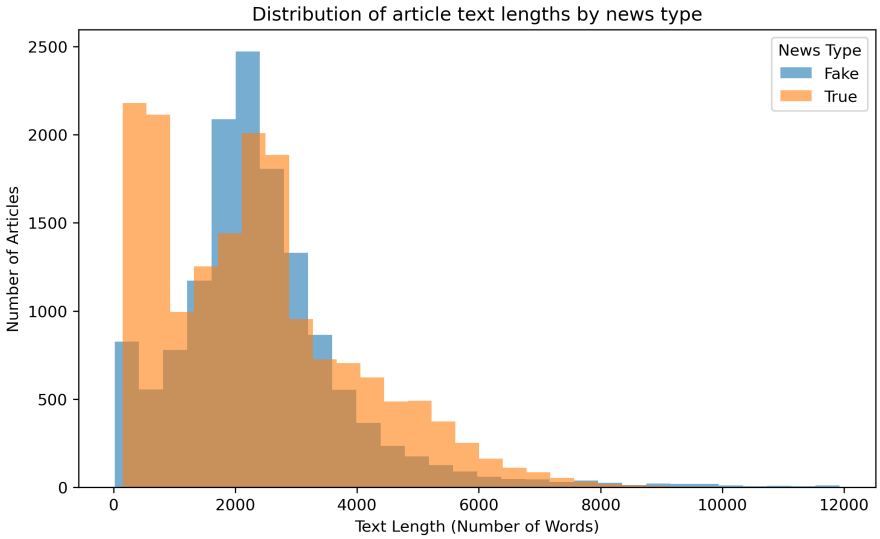

Figure 7: Distribution of article text lengths for fake and true news articles

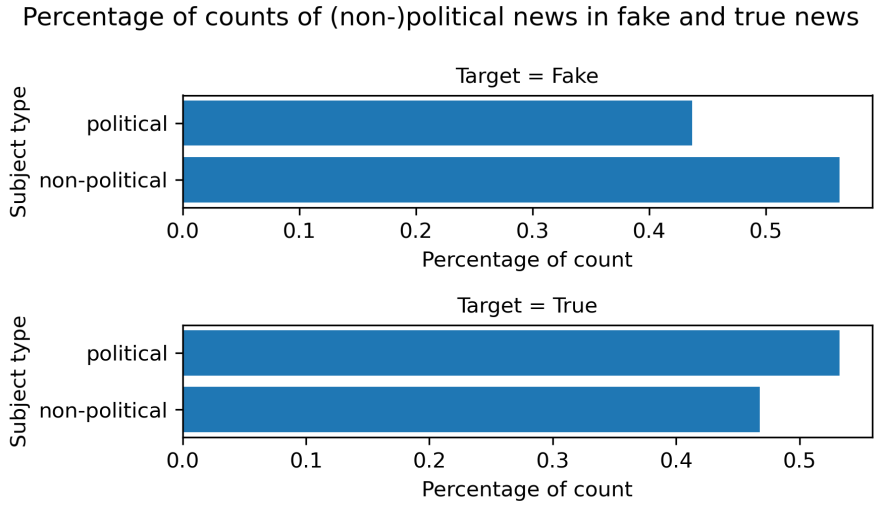

Figure 8: Percentage of counts of news subjects in fake vs. true news articles

in predicting whether news is true. Therefore, the feature `subject` (political vs non-political) is included in the classification model below.

4.2 Classfication Modeling Analysis

4.2.1 Dummy Classifier

Before evaluation our model, it is important to establish a baseline for comparison. The Dummy classifier acts as this baseline by applying a simple strategy to classify observations rather than learn from the data. In this case, our Dummy classifier always predicts the majority class, using the "most_frequent" strategy. The goal later on is to build a classification model that does well compared to this Dummy.

4.2.2 Naive Bayes Classifier

`MultinomialNB` is used as the estimator to deal with integer counts data, after proper preprocessing of the features. Two textual features (`title` and `text`) and a categorical feature `subject` with only two levels (*political* and *non-political*) are included in modelling. The `date` feature is dropped due to its irrelevance to the target. Genereally speaking, when the news is published is random information and has little to do with whether the news is fake or true.

As a first step, the features should be preprocessed. One-Hot encoding is applied to the categorical feature `subject` with `drop="if_binary"` to include only one column for this binary feature. For the two textual features, `title` and `text`, they are first transformed by `lambda x: np.ravel(x)` to be one-dimensional arrays. After being flattened, they are then passed to `CountVectorizer` to extract the words and number of occurrences from these two features. Note that `CountVectorizer` needs to be created twice, one for each textual feature.

Next, a machine learning pipeline `pipe` is created, with preprocessing and the Naive Bayes estimator. Before fitting the model to the training set, hyperparameter tuning is carried out to boost classification performance. There are 3 hyperparameters being considered: - `max_features` for the title `CountVectorizer` (randomly sampled from 1-200) - `max_features` for the text `CountVectorizer` (randomly sampled from 1-5000), - `alpha` of `MultinomialNB` (randomly sampled from 10^{-7}-10^{0}). Randomized search with cross-validation selects the best combination of these parameters.

Note that `RandomizedSearchCV` runs very slowly in this case. To reduce computation time, `n_iter` is set to only 10. Albeit doing only 10 searches, the best model found already performs very well, as we can see from the CV score and test scores below.

5 Results

As a baseline, dummy classifier is first fitted to get the test accuracy.

Using most frequent dummy classifier, the test score is low (only 0.538) due to a relatively balanced dataset. Now compare the Naive Bayes model with the dummy classifier.

After fitting the Naive Bayes model to the training set, the best hyperparameters and CV score can be extracted from the model. The best values of all 3 hyperparameters are in the middle of the search ranges instead of being on the edge, so the search ranges are specified appropriately. Under this model, the training score is 0.955. The best CV score is 0.954, and the test score is 0.954. Since the train, CV and test score are close to one another and are both high, underfitting or overfitting is not a concern here. Compared to dummy classifier, this Naive Bayes model performs a lot better in terms of prediction accuracy, increasing the test score from 0.538 to 0.954.

The Naive Bayes model demonstrates strong performance in distinguishing between fake and real news articles based on their textual characteristics, as shown in the word clouds (Figure 2 and Figure 3) which reveal distinct vocabulary patterns between fake and real news.

5.1 Model Evaluation

Below are the visualizations of the classification results on the test set, as a model performance assessment.

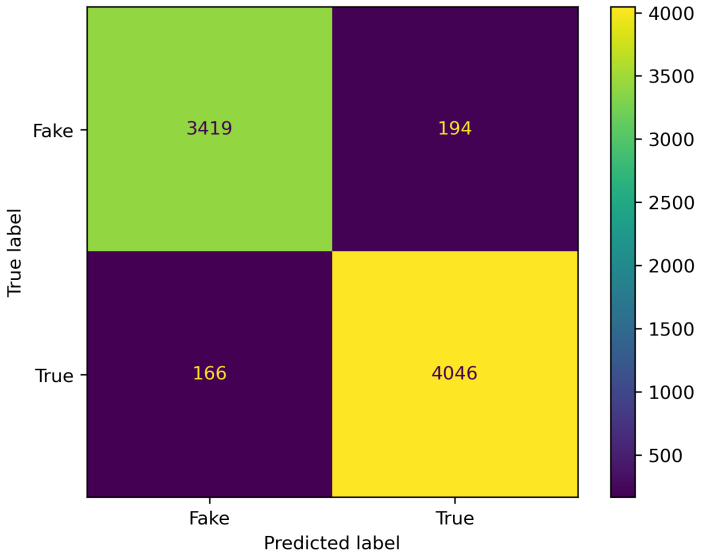

Figure 9: Confusion matrix for Naive Bayes classifier on test set

	precision	recall	f1-score	support
Fake	0.95	0.95	0.95	3613
True	0.95	0.96	0.96	4212
accuracy			0.95	7825
macro avg	0.95	0.95	0.95	7825
weighted avg	0.95	0.95	0.95	7825

The model classifies both true and fake news correctly most of the time, indicated by a high true postive and true negative rate. Both false positve and false negative rates are low, since there are few misclassifications. Therefore, not only is the model accuracy high, but the model is also robust in correctly detecting both true and fake news, with few misclassifications.

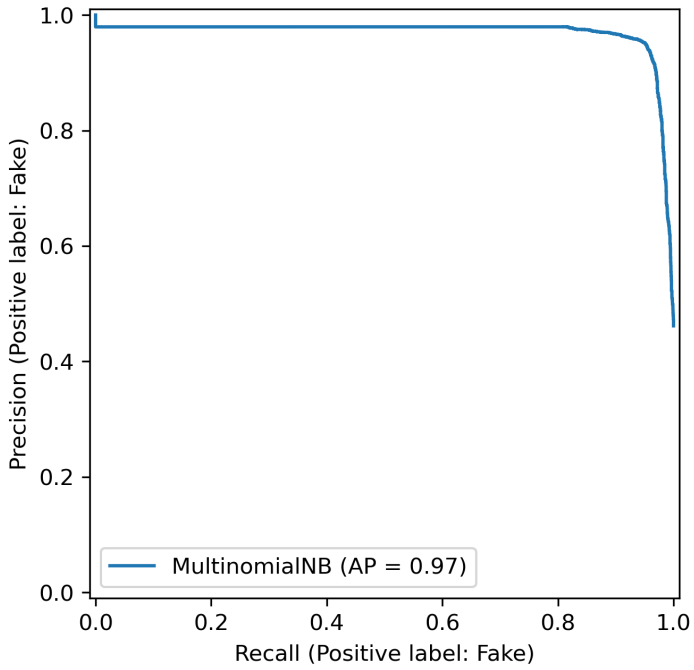

Figure 10: Precision-Recall curve for Naive Bayes classifier on test set

The AP score, as a summary of the PR curve, is 0.97. That fact that the AP score is very close to 1 suggests that our Naive Bayes model's performance is excellent.

The AUC here is 0.98, which is much higher than the baseline 0.5, further indicating the satisfactory performance of the Naive Bayes model.

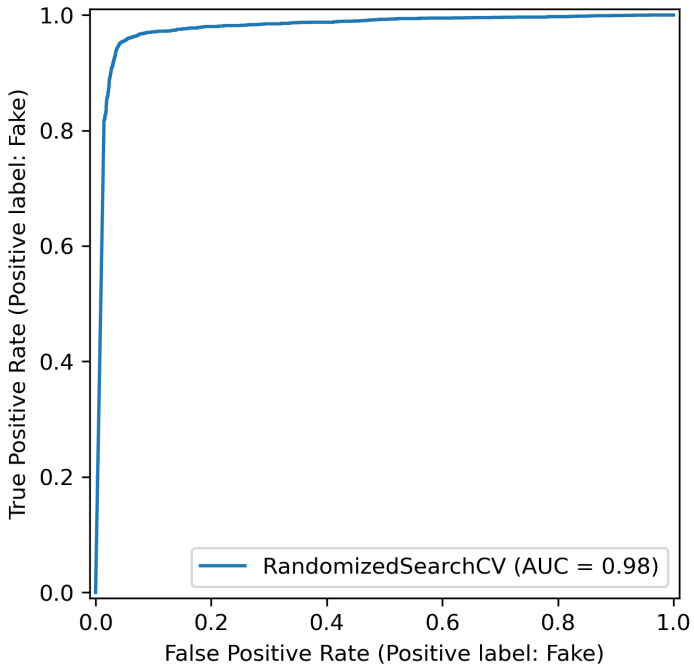

Figure 11: ROC Curve for Naive Bayes classifier on test set

6 Discussion

6.1 Results Summary

The Naive Bayes model performs decently well due to high accuracy, high true positive rate, high true negative rate, low false postive rate, low false negative rate, high AP score and high AUC value. Apart from these exceptional evaluation metrics, this model also does not throw major concerns such as underfitting or overfitting problems. The training score (0.955), the best CV score (0.954), and test score (0.954) are consistently high and close to one another, so there is no noticeable underfitting or overfitting issue. To summarize, compared to the baseline model (dummy classifier), this Naive Bayes model exhibits major improvements, returning satisfactory results.

Our findings are partly aligned with expectations. According to the nature of political news, it is expected that there is more political news under the fake news category. However, in this dataset, there is only 2% more political news in fake news compared to true news. This marginal difference is not expected.

Despite this, the Naive Bayes model generally yields high performance, even if it is simple. So, using `MultinomialNB` with hyperparameter tuning gives an expected high prediction performance. Moreover, the original counts of words are used instead of the binary information

presence/absence, so there is no loss of information in modelling. Therefore, the satisfactory result is as expected.

6.2 Impact of these Findings

This Naive Bayes model could potentially be deployed and used for news classification, which may help news readers identify fake news and avoid accepting false information. Secondly, this news classification model could also assist news agents screen and select only the true news to publish, avoiding losing reputation for articulating fake news. Lastly, this model could inspire further improvements of news classification models, acting as a good starting point for future model building.

Specific to this particular model, two future questions could be asked. First, the percentage of counts of political news is only marginally different between true and fake news. So, it is worth exploring whether dropping the `subject` feature will increase or decrease the model performance. This can be done by comparing the two models with and without the `subject` feature. Second, what other possible features can be used or engineered to further improve the model performance? This can be explored by asking for suggestions from human experts with domain knowledge. So far only the Naive Bayes model is used, which is a simple and naive model. Not restricted to this model anymore, there are other questions that can be raised. For example, are there any other more advanced models that can deal with integer count data and have more satisfying results?

6.3 Limitations of Analysis

All news are classified as political and non-political in the analysis, which is crude and leaves space for improvement. More detailed `subject` categories can be applied to improve model performance. Besides that, all the collected data we have do not cover all types of news in the world. The topics included in the dataset are rather quite limited. So, this model could do well when seeing similar news, but could fail when given a totally unfamiliar news topic. So enriching the training data is essential to deploy the model in the wild.

7 Conclusion

The machine learning approach using Naive Bayes classification shows promising results for fake news detection. The model's high accuracy and balanced performance across both classes suggest it could be a valuable tool in combating misinformation. However, the model's performance may be limited to the types of news articles present in the training data, and further validation on diverse news sources would be beneficial for real-world deployment.

References

Allcott, Hunt, and Matthew Gentzkow. 2017. "Social Media and Fake News in the 2016 Election." *Journal of Economic Perspectives* 31 (2): 211–36. https://doi.org/10.1257/jep.31.2.211.

Bilodeau, Howard, and Aisha Khalid. 2024. "The Spread of Misinformation: A Multivariate Analysis of the Relationship Between Individual Characteristics and Fact-Checking Behaviours of Canadians."

Bisaillon, Clément. 2023. "Fake and Real News Dataset." /url%7Bhttps://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset%7D.

Canada, Service. 2024. "Online Disinformation: Learn about It." /url%7Bhttps://www.canada.ca/en/campaign/online-disinformation/learn-about-it.html%7D.

Pennycook, Gordon, and David G. Rand. 2021. "The Psychology of Fake News." *Trends in Cognitive Sciences* 25 (5): 388–402. https://doi.org/10.1016/j.tics.2021.02.007.