Enhancing Fake News Detection from News Headlines through Advanced NLP Techniques

Chih-Hsuan Lin, Po-Chih Chang, Yen-Hung Ho
Virginia Tech

1 PROJECT IDEA

Fake news has always been a problem in modern social software. False information can cause serious social harm, and even cause financial or life damage to users. We found that although Natural Language Processing (NLP) can effectively distinguish fake news, most of them use news text for feature extraction, while most of the fake news in social software only display headlines. We hope to find an effective way to categorize news headlines using NLP technology, through which we can detect fake news in social software more efficiently.

We plan to develop a fake news detection system by applying various natural language processing techniques we have learned in class. Using a dataset from Kaggle [2] with 23,502 fake news articles and 21,417 real news articles, we will explore different methods to distinguish fake news from real news. We will use the headlines and text of the articles to train our models.

2 MOTIVATION

Fake news spreads quickly on social media and can have serious consequences [1] [4] [5]. Detecting fake news is important to keeping people informed with accurate information. When we look for sources of fake news, we find that on Twitter, Facebook, or reddit, news-type posts usually only show a link to the news with the headline or some sort of summary. We have to click on the link to get to the news article. This makes us think that if we use news headlines to train the model, we can instantly detect fake news in social software and prevent users from being misled by fake news. Our project allows us to apply what we've learned in our social media analysis course to a real-world problem.

3 DATASET

We will use a fake news detection dataset on Kaggle, which includes headlines and full text of news articles labeled as fake or real. The dataset consists of two separate files: one for fake news and one for real news. Each file contains fields such as title, body, subject, and date. The title field contains the headline of the article, and the text field contains the main content. The subject field indicates the category or topic of the article, and the date field indicates when the article was published. This dataset is suitable for our project because it provides enough data for training and testing different models and contains the necessary text fields for our analysis.

Table 1: Fake News Example

Field	Content
Title	Donald Trump Sends Out Embarrassing New Year's Eve Message
Text	Donald Trump just couldn't wish all Americans a Happy New Year and...
Subject	News
Date	2017-12-31
Tag	Fake

Table 2: Real News Example

Field	Content
Title	FBI officials said Clinton 'has to win' race to White House: NYT
Text	(Reuters) - Senior FBI officials who helped probe Donald Trump's 2016 presidential campaign told a c...
Subject	politicsNews
Date	2017-12-13
Tag	Real

4 APPROACHES

We will gradually analyze the dataset in three steps. First, we will analyze the dataset through Named Entity Recognition (NER) analysis and topic modeling to find out whether the distribution of topics in the news is overly skewed towards political news, which will help us to improve the problem of skewed topics by capturing the keywords and replacing them with generic text through NER subsitution later.

In the second part, we apply the different feature extraction methods we learned in class and try to extract different semantic features from the news text and headlines, such as term frequency-inverse document frequency (TF-IDF) and Bag of Words (BoW) for frequency-based embeddings, Word2Vec and GloVe for sentiment embeddings, and more complex Universal Sentence Encoder (USE), etc., and use these features to train the model and try to let the model perform the prediction.

In the third part, we aim to address the limitations observed in the first part by mitigating biases in the data set. Using the results from topic modeling, we replace biased topics - such as political figures or entities that dominate the data - with generic tokens. This substitution allows us to evaluate whether neutralizing biased entities can improve the model's ability to generalize across topics and improve prediction accuracy.

4.1 Data Preprocessing and Exploration

The first phase of our analysis focuses on understanding the inherent characteristics and potential biases in our data set through two

Conference'17, July 2017, Washington, DC, USA

main approaches: named entity distribution analysis and sentiment analysis.

4.1.1 Named Entity Recognition and Topic Distribution. We use spaCy's NER model to extract named entities from both fake and real news headlines. This process helps to identify the frequency and distribution of different entity types (e.g. PERSON, ORG, GPE) across both categories. For topic distribution analysis, we use Latent Dirichlet Allocation to generate topic clusters and visualize them through word clouds. This approach is particularly important because preliminary observations suggest potential topic imbalances-for example, fake news may be disproportionately concentrated in political content compared to real news.

The process involves:

- Applying spaCy's pre-trained model for NER
- Aggregating entity frequencies and categorizing by news type
- Implementing Latent Dirichlet allocation (LDA) using gensim with the following parameters
- Generating interactive word clouds using WorldCloud library to visualize topic distributions

4.1.2 Sentiment Analysis. We implement Valence Aware Dictionary and sEntiment Reasoner (VADER) sentiment analysis to examine the emotional tone differences between fake and real news headlines. VADER is particularly suitable for this task as it is specifically attuned to sentiments expressed in social media and news headlines. The analysis produces four key metrics for each headline:

- Positive sentiment score
- Negative sentiment score
- Neutral sentiment score

4.2 Feature Extraction and Classification

Our approach to fake news detection combines different text representation techniques with multiple classification methods to evaluate their comparative effectiveness. We implement traditional NLP feature extraction methods.

4.2.1 Feature Extraction Methods. Our text representation techniques include both traditional and advanced methods for capturing different aspects of news headlines. The traditional methods start with BoW, which implements a frequency-based vector representation using sklearn's CountVectorizer with default parameters. We then use TF-IDF to capture the importance of terms while taking into account the frequency of documents. For word embeddings, we use pre-trained Word2Vec embeddings from Google News (300d) with mean pooling for sentence representation, followed by pre-trained GloVe embeddings with mean pooling.

For advanced sentence embeddings, we implement two sophisticated approaches. First, we use Google's USE library to generate 512-dimensional embeddings that capture semantic relationships.

4.2.2 Classification Methods. Our evaluation encompasses several classification approaches, including traditional machine learning methods mentioned in the class. The machine learning approaches include Logistic Regression (LR), Random Forest (RF), Decision Tree (DT) and Support Vector Machine (SVM).

4.2.3 Evaluation Strategy. Our evaluation strategy uses a systematic approach for each combination of feature extraction method and classifier. We perform 5-fold cross-validation and compute standard metrics such as accuracy, precision, and recall.

This comprehensive evaluation allows us to identify the most effective combination of feature extraction and classification methods for headline-based fake news detection. The results are compared to determine which approach provides better performance for this specific task.

4.3 Model Enhancement through Entity Neutralization

To further explore and address biases in model predictions, we extended our approach by introducing the FakeNewsNet dataset [3] to assess model generalizability. This additional dataset allowed us to test whether a model trained on the original dataset could effectively predict fake and real news in a new domain. Our evaluation included assessing baseline performance as well as the impact of entity neutralization.

Entity neutralization was performed to systematically reduce the influence of certain named entities that could skew predictions due to their overrepresentation in the training data. Using Named Entity Recognition (NER), we identified and replaced five primary entity types in the news headlines: `PERSON`, `ORGANIZATION`, `DATE`, `TIME`, and `GPE`. In addition, we adjusted the replacement dictionary to remove high-frequency political terms such as "Donald Trump" and "Hillary Clinton". This preprocessing step resulted in more generalized datasets that allowed us to test the effect of neutralized inputs on model performance.

Examples of entity replacement including:

- `"Donald Trump"` → `"PERSON"`
- `"FBI"` → `"ORGANIZATION"`
- `"December 2023"` → `"DATE"`

After preprocessing, we retrained the models using different text representation methods such as BoW, TF-IDF, and embeddings such as Word2Vec and GloVe. These models were then tested on both the original and FakeNewsNet datasets to evaluate the generalizability and impact of entity neutralization.

Preliminary results indicated that entity neutralization improved model accuracy by approximately 10% on the FakeNewsNet dataset compared to the baseline. However, challenges remained, particularly in recall for fake news detection, suggesting that additional techniques such as data balancing or domain adaptation may be necessary for further improvement. Overall, this approach highlights the potential of NER-based preprocessing to improve model robustness in cross-dataset scenarios.

5 RESULTS

5.1 Dataset exploration and Visualization

5.1.1 NER Analysis. The NER analysis was conducted to uncover biases in the dataset and to better understand the characteristics of fake and real news. This analysis aimed to identify differences in the frequency and types of entities mentioned in each category, as well as the underlying topics that dominate the articles.

NER was performed (Figure 1) to extract and quantify entities such as PERSON, ORG, GPE, and others. The results show remarkable differences in entity types between fake and real news. For fake news, the most frequently mentioned entities include PERSON (158,015 mentions), ORG (86,539 mentions), and GPE (74,139 mentions). In addition, fake news has a high frequency of entities such as NORP (nationalities/religions) and DATE, suggesting a strong focus on events related to political figures and timelines.

In contrast, real news shows a different distribution of entities. The most common entities in real news are GPE (137,812 mentions), PERSON (112,625 mentions) and ORG (106,578 mentions). Compared to fake news, real news tends to have more mentions of GPE, indicating a stronger focus on geographic references.

While fake news tends to highlight individuals and politically charged issues to attract attention, real news has a broader coverage of places and institutions, indicating a more balanced and authoritative reporting style.

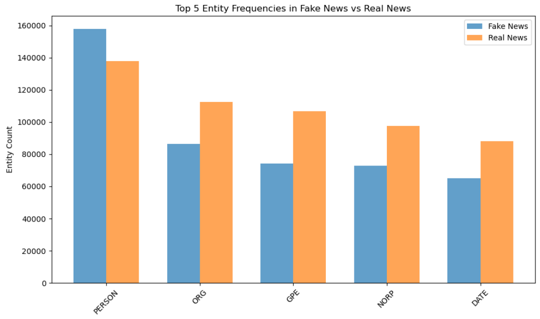

Figure 1: Top 5 Entity Frequencies in Fake News vs Real News.

5.1.2 Topic Distribution. Topic modeling using Latent Dirichlet Allocation (LDA) identified five dominant topics in each category:

Fake News Topics:

- Police violence and social issues: *'video', 'police', 'gun', 'man', 'people'*.
- Education and law: *'school', 'vote', 'law', 'state'*.
- Hillary Clinton and Donald Trump: *'hillary', 'clinton', 'trump'*.
- Russian interference: *'russian', 'email', 'fbi', 'news'*.
- National politics and economic issues: *'america', 'country', 'million'*.

Real News Topics:

- European politics and taxation: *'minister', 'election', 'tax', 'party'*.
- International conflicts and government forces: *'kill', 'military', 'attack', 'force'*.
- Foreign policy and nuclear issues: *'official', 'korea', 'china', 'united'*.
- U.S. politics and campaigns: *'white', 'house', 'republican', 'campaign'*.
- Federal legislation and governance: *'senate', 'court', 'law', 'state'*.

These topics reflect a notable bias: fake news articles often focus on sensational topics like political scandals, while real news covers broader geopolitical and policy issues.

5.1.3 Word Clouds. To visualize the common words in fake and real news, word clouds were generated (Figure 2). Our key observation on fake news is that words like *trump*, *say*, and *hillary clinton* dominate, reflecting a tendency toward sensationalism and political bias. For real news, words like *united states*, *white house*, and *government* are more prominent, indicating a formal and broader coverage of national and international issues.

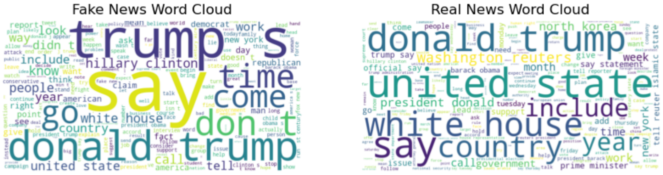

Figure 2: Word clouds for fake and real news articles.

5.1.4 VADER analysis. Sentiment analysis was conducted using the VADER sentiment analyzer to compare the emotional tone of fake and real news articles. This analysis evaluates four key metrics: negative (neg), neutral (neu), positive (pos), and compound sentiment scores. Furthermore, NRC-VAD scores were integrated with VADER metrics and other features (e.g., Named Entity Recognition) to assess predictive performance and consistency. The VADER analysis revealed notable differences between fake and real news. Fake news exhibited more extreme negative (*neg*) and compound scores, indicating a tendency towards polarizing or sensational language. Real news, by contrast, displayed higher neutral (*neu*) sentiment scores, reflecting its objective and balanced reporting style. Statistical results of the T test confirmed these differences as statistically significant ($p < 0.01$ for all metrics). For example:

- *neg*: T-statistic = 26.97, p = 6.68e-159.
- *neu*: T-statistic = -45.74, p = 0.00.
- *pos*: T-statistic = 13.51, p = 1.66e-41.

Visual Patterns Figures 3 and 4 illustrate the differences in sentiment. The first difference is **Average Sentiment Scores:** Fake news scored slightly higher on negative sentiments, while real news scored higher on positive and neutral sentiments (Figure 3). The second difference is **Boxplot of Compound Sentiment Scores:** For fake news, the median compound score is close to 0, indicating a general trend of neutral sentiment. However, the distribution is wide, ranging from extremely negative (-1.0) to extremely positive (1.0), suggesting that fake news often employs more extreme sentiment to attract attention. In contrast, real news has a slightly positive median compound score, reflecting a tendency toward a more positive tone. Its distribution is also wide but with slightly shorter whiskers, indicating a more consistent and stable sentiment compared to fake news (Figure 4). The compound scores for fake news are more polarized, showing a tendency to adopt stronger emotional expressions, positive or negative. Real news, on the other hand, is more concentrated around neutral or slightly positive scores, consistent with its goal of balanced and objective reporting. These differences suggest that the sentiment distribution could serve as a key feature for automated fake news detection, leveraging the more extreme nature of fake news sentiments as a distinguishing factor.

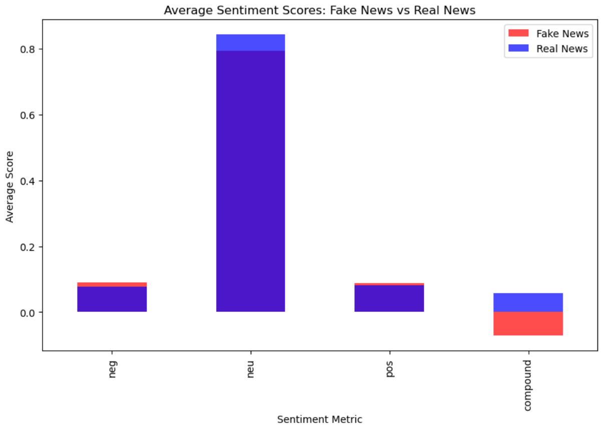

Figure 3: Average sentiment scores for fake and real news articles.

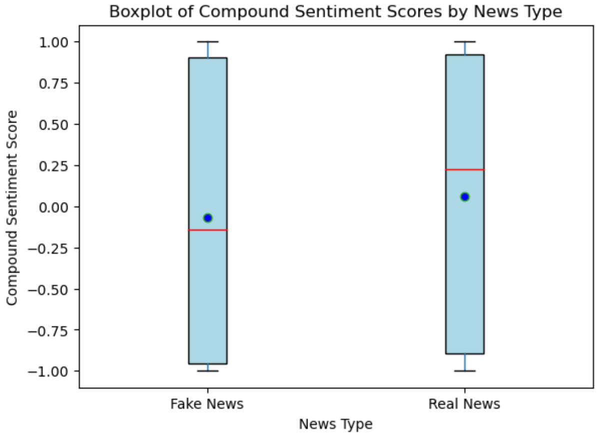

Figure 4: Boxplot of compound sentiment scores for fake and real news articles.

5.1.5 NRC-VAD scores. The NRC-VAD scores were analyzed to examine the emotional tone, intensity, and authority present in fake and real news articles. Three dimensions were calculated: valence, arousal, and dominance. Valence measures the positivity or negativity of the text, arousal indicates the level of emotional intensity, and dominance reflects the level of control or authority conveyed. The analysis reveals notable differences between fake and real news. Real news articles showed slightly higher valence scores, averaging 0.599, compared to fake news, which averaged 0.602. This suggests a more positive emotional tone in the real news. The arousal scores were slightly higher for fake news (0.478 for fake news versus 0.467 for real news), indicating a slightly higher emotional intensity in fake news headlines. Dominance scores were consistently higher in real news, averaging 0.604 compared to 0.549 for fake news, reflecting a more assertive and authoritative language style in real news articles. These findings highlight key linguistic differences between fake and real news, with real news tending to adopt a more balanced, credible tone, and fake news leaning towards more emotionally charged expressions. The NRC-VAD dimensions further demonstrate their potential utility as features for distinguishing fake news from real news in automated detection systems.

5.2 Model performance

We first explored the different embedding methods and tested their ability to classify the fake news data set. We used three types of methods: frequency-based embedding, such as BoW and TF-IDF; pre-trained embedding model, such as GloVe and Word2Vec; advanced embedding model Universal Sentence Encoder and NRC-VAD scores as an additional feature.

5.2.1 Frequency-based embedding. For the BoW embedding, we employed feature selection using the chi-square test to identify the most discriminative features, selecting the top 100 features to reduce dimensionality and prevent overfitting. As shown in Table 5, both BoW and TF-IDF achieved comparable performance across different classifiers. The BoW approach achieved the highest accuracy of 84.70% with SVM, demonstrating strong precision for true news (0.92) and balanced recall rates (0.85/0.84) for both fake and true news categories.

TF-IDF vectorization showed similar performance patterns, with the highest accuracy of 84.41% also achieved using SVM. Notably, TF-IDF consistently demonstrated higher precision for true news (0.94) across most classifiers, while maintaining strong recall rates for fake news detection (up to 0.95 with Logistic Regression). These results suggest that both frequency-based embedding methods provide effective baseline performance for headline-based fake news detection, with slight advantages in different evaluation metrics.

5.2.2 Pre-trained embedding. We evaluated two popular pre-trained word embeddings: Word2Vec trained on Google News and GloVe trained on Common Crawl. As shown in Table 5, both pre-trained embeddings demonstrated substantial improvements over frequency-based methods across most classifiers. Word2Vec achieved the highest overall performance, reaching 93.77% accuracy with SVM classification. This combination also showed excellent balanced performance with precision (0.93/0.95) and recall (0.94/0.93) for both fake and true news categories.

GloVe embeddings performed similarly well, with its best performance of 92.39% accuracy also achieved using SVM. The consistency between Word2Vec and GloVe results suggests that pre-trained embeddings effectively capture semantic relationships crucial for fake news detection. Notably, while both embedding types showed strong performance with SVM and Random Forest classifiers, Decision Tree classifiers performed significantly worse (80.74% for Word2Vec, 82.35% for GloVe), indicating that the semantic space created by these embeddings may not be well-suited for decision tree partitioning.

5.2.3 Universal Sentence Encoder. For sentence-level embeddings, we utilized Google's USE, which captures semantic relationships at the sentence level rather than individual words. As shown in Table 5, USE demonstrated superior performance compared to both frequency-based and pre-trained word embedding methods. The highest accuracy of 94.30% was achieved using SVM classification, with exceptionally balanced performance metrics showing precision of 0.93/0.95 for fake/true news and recall of 0.95/0.94 respectively.

The strong performance was consistent across most classifiers, with Random Forest achieving 92.31% accuracy and Logistic Regression reaching 91.15%. Similar to other embedding methods,

Table 3: Results of embeddings in classifying original dataset

Feature	Model	Accuracy (%)	Precision (Fake/True)	Recall (Fake/True)
BoW	Logistic Regression	84.52	0.78 / 0.93	0.93 / 0.77
BoW	Decision Tree	84.25	0.78 / 0.93	0.93 / 0.76
BoW	Random Forest	84.61	0.78 / 0.93	0.85 / 0.84
BoW	Support Vector Machine	84.70	0.79 / 0.92	0.85 / 0.84
TF-IDF	Logistic Regression	83.87	0.77 / 0.94	0.95 / 0.74
TF-IDF	Decision Tree	83.25	0.77 / 0.92	0.84 / 0.82
TF-IDF	Random Forest	84.07	0.77 / 0.94	0.94 / 0.75
TF-IDF	Support Vector Machine	84.41	0.77 / 0.94	0.95 / 0.75
Word2Vec	Logistic Regression	90.59	0.90 / 0.91	0.91 / 0.91
Word2Vec	Decision Tree	80.74	0.81 / 0.80	0.77 / 0.84
Word2Vec	Random Forest	91.18	0.89 / 0.93	0.93 / 0.90
Word2Vec	Support Vector Machine	93.77	0.93 / 0.95	0.94 / 0.93
GloVe	Logistic Regression	90.05	0.89 / 0.91	0.90 / 0.90
GloVe	Decision Tree	82.35	0.83 / 0.82	0.79 / 0.85
GloVe	Random Forest	90.90	0.89 / 0.92	0.92 / 0.90
GloVe	Support Vector Machine	92.39	0.91 / 0.93	0.93 / 0.92
USE	Logistic Regression	91.15	0.90 / 0.92	0.92 / 0.91
USE	Decision Tree	84.65	0.86 / 0.84	0.81 / 0.88
USE	Random Forest	92.31	0.92 / 0.93	0.92 / 0.92
USE	Support Vector Machine	94.30	0.93 / 0.95	0.95 / 0.94
NRC-VAD	Logistic Regression.	64.35	0.63 / 0.65	0.59 / 0.69
NRC-VAD	Decision Tree	65.73	0.65 / 0.66	0.60 / 0.71
NRC-VAD	Random Forest	70.35	0.70 / 0.71	0.65 / 0.75
NRC-VAD	Support Vector Machine	65.00	0.65 / 0.65	0.58 / 0.71

Decision Tree classifier showed relatively lower performance at 84.65%, though this was still higher than its performance with frequency-based embeddings. The consistent high performance across different classifiers suggests that USE's sentence-level semantic representations are particularly effective for fake news detection from headlines.

5.2.4 NRC-VAD Sentiment Analysis. To investigate the role of emotional content in fake news detection, we employed the NRC Valence, Arousal, and Dominance (NRC-VAD) lexicon for sentiment analysis. As presented in Table 5, the sentiment-based features alone achieved moderate performance levels, with the highest accuracy of 70.35% obtained using Random Forest classification. The results show a consistent pattern across all classifiers where true news detection slightly outperforms fake news detection, as evidenced by the higher recall rates for true news (ranging from 0.69 to 0.75).

While these accuracy scores are notably lower than those achieved by semantic embedding methods, they provide valuable insights into the emotional characteristics of fake and true news headlines. The performance suggests that while sentiment features alone are not sufficient for reliable fake news detection, they could potentially complement other features as part of a more comprehensive approach.

5.2.5 Cross-Dataset Performance. To evaluate the generalizability of our models, we performed a cross-dataset validation using the FakeNewsNet dataset. As shown in Table 4, the performance metrics revealed significant challenges in transferring learned patterns across different fake news datasets. Models trained on the original dataset showed a significant drop in performance when tested on FakeNewsNet, with baseline accuracies ranging from 35% to 70%, depending on the preprocessing method and feature extraction technique. This drop highlights the challenges posed by domain differences and data imbalance. FakeNewsNet, with its disproportionately large number of real news samples (17,000) compared to fake news samples (5,700), strongly influenced model performance.

These results highlight a significant challenge in fake news detection: models trained on one dataset may not effectively generalize to other sources of fake news. This suggests that fake news patterns are highly context-dependent and may require more sophisticated approaches or domain adaptation techniques for effective cross-dataset application.

While accuracy for FakeNewsNet was often higher than 60% for Word2Vec, GloVe and USE, this metric is misleading due to poor recall for fake news, as many models prioritized detection of real news at the expense of identifying fake news. In contrast, frequency-based methods such as BoW and TF-IDF showed better recall for fake news, making them more suitable in this context where identifying fake news is the primary objective.

5.3 Model Enhancement Result

In this section, we present the results of our model improvement experiments, focusing on the impact of entity neutralization (NER preprocessing) and cross-dataset generalization. Our analysis included training the models on the original dataset and testing their performance on both the original and FakeNewsNet datasets. The results were evaluated using accuracy, precision, and recall metrics, with particular emphasis on recall for fake news detection.

Table 4: Results of embeddings in classifying FakeNewsNet dataset

Feature	Model	Accuracy (%)	Precision (Fake/True)	Recall (Fake/True)
BoW	Logistic Regression	36.13	0.25 / 0.75	0.78 / 0.22
BoW	Decision Tree	35.43	0.25 / 0.76	0.80 / 0.21
BoW	Random Forest	35.46	0.25 / 0.76	0.80 / 0.21
BoW	Support Vector Machine	35.48	0.25 / 0.75	0.79 / 0.21
TF-IDF	Logistic Regression	34.43	0.25 / 0.76	0.82 / 0.19
TF-IDF	Decision Tree	33.09	0.25 / 0.74	0.82 / 0.17
TF-IDF	Random Forest	33.26	0.25 / 0.74	0.82 / 0.17
TF-IDF	Support Vector Machine	34.49	0.25 / 0.76	0.81 / 0.19
Word2Vec	Logistic Regression	68.30	0.24 / 0.75	0.13 / 0.87
Word2Vec	Decision Tree	59.27	0.23 / 0.74	0.27 / 0.70
Word2Vec	Random Forest	63.56	0.22 / 0.74	0.18 / 0.78
Word2Vec	Support Vector Machine	61.76	0.22 / 0.73	0.21 / 0.75
GloVe	Logistic Regression	69.59	0.20 / 0.75	0.07 / 0.90
GloVe	Decision Tree	57.74	0.25 / 0.75	0.36 / 0.65
GloVe	Random Forest	66.55	0.22 / 0.75	0.14 / 0.84
GloVe	Support Vector Machine	62.06	0.23 / 0.75	0.22 / 0.75
USE	Logistic Regression	68.06	0.22 / 0.75	0.11 / 0.87
USE	Decision Tree	59.69	0.24 / 0.75	0.28 / 0.70
USE	Random Forest	64.93	0.21 / 0.74	0.15 / 0.81
USE	Support Vector Machine	66.86	0.23 / 0.75	0.14 / 0.84
NRC-VAD	Logistic Regression.	65.79	0.26 / 0.75	0.20 / 0.81
NRC-VAD	Decision Tree	50.91	0.25 / 0.75	0.48 / 0.52
NRC-VAD	Random Forest	56.00	0.24 / 0.75	0.35 / 0.63
NRC-VAD	Support Vector Machine	62.77	0.24 / 0.75	0.23 / 0.76

NER had a mixed effect on model performance. A comparison of model performance between the original dataset 5 and FakeNewsNet 6 reveals several key insights into the effectiveness of NER processing.

For frequency-based methods (BoW and TF-IDF), NER showed significant improvements when classifying FakeNewsNet data, increasing accuracy by about 10% (from 35% to 45%). This suggests that removing named entities helps these models focus on more generalizable content patterns rather than dataset-specific entities. NER processing significantly reduced the accuracy of the GloVe embedding model. NER processing had minimal impact on their FakeNewsNet performance, with differences of only 2% between processed and unprocessed versions. The NRC-VAD sentiment features showed the most consistent performance across both datasets, suggesting that emotional content may be a more stable indicator of news authenticity than specific textual features. However, its overall accuracy was lower than other methods in the original dataset, indicating that sentiment alone is not sufficient for reliable fake news detection.

Overall, the results indicate that while NER preprocessing can improve generalizability for certain methods, it requires careful consideration of the trade-offs between recall for fake news and real news. Frequency-based techniques like TF-IDF without NER preprocessing may offer the most robust and balanced performance for fake news detection.

6 CONCLUSION

Our comprehensive study on fake news detection from headlines using various NLP techniques has yielded several important findings and insights. First, our experiments show that advanced embedding methods, in particular USE, achieve superior performance on the original dataset, reaching 94.30% accuracy with SVM classification. However, the significant performance degradation observed in cross-dataset testing (dropping to 66.86% accuracy) highlights a critical challenge in developing generalizable fake news detection systems.

Our entity neutralization experiments provide valuable insights into the role of named entities in fake news detection. While NER preprocessing improved cross-dataset performance for frequency-based methods by about 10%, its impact varied significantly across different embedding approaches. This suggests that entity-specific information, while potentially introducing bias, also carries important signals for fake news detection that need to be carefully balanced.

The sentiment analysis results, particularly the NRC-VAD scores, reveal distinct emotional patterns between fake and real news headlines. Although sentiment features alone proved insufficient for reliable classification, they showed the most consistent performance across datasets, suggesting their potential value as complementary features in a more comprehensive detection system.

These findings suggest several directions for future work. First, developing more sophisticated domain adaptation techniques could help address the significant performance gap in cross-dataset scenarios. Second, investigating hybrid approaches that combine the robustness of frequency-based methods with the sophisticated semantic understanding of advanced embeddings might yield more generalizable models. Finally, exploring ways to effectively integrate sentiment analysis with other features while maintaining cross-domain performance could lead to more reliable fake news detection systems.

REFERENCES

[1] Hunt Allcott and Matthew Gentzkow. 2017. Social media and fake news in the 2016 election. *Journal of economic perspectives* 31, 2 (2017), 211–236.

[2] clmentbisaillon. 2020. Fake and real news dataset. https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset. (2020).

[3] Aleksei Golovin. 2022. Fake News. https://www.kaggle.com/datasets/algord/fake-news. (2022).

[4] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter* 19, 1 (2017), 22–36.

[5] Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *science* 359, 6380 (2018), 1146–1151.

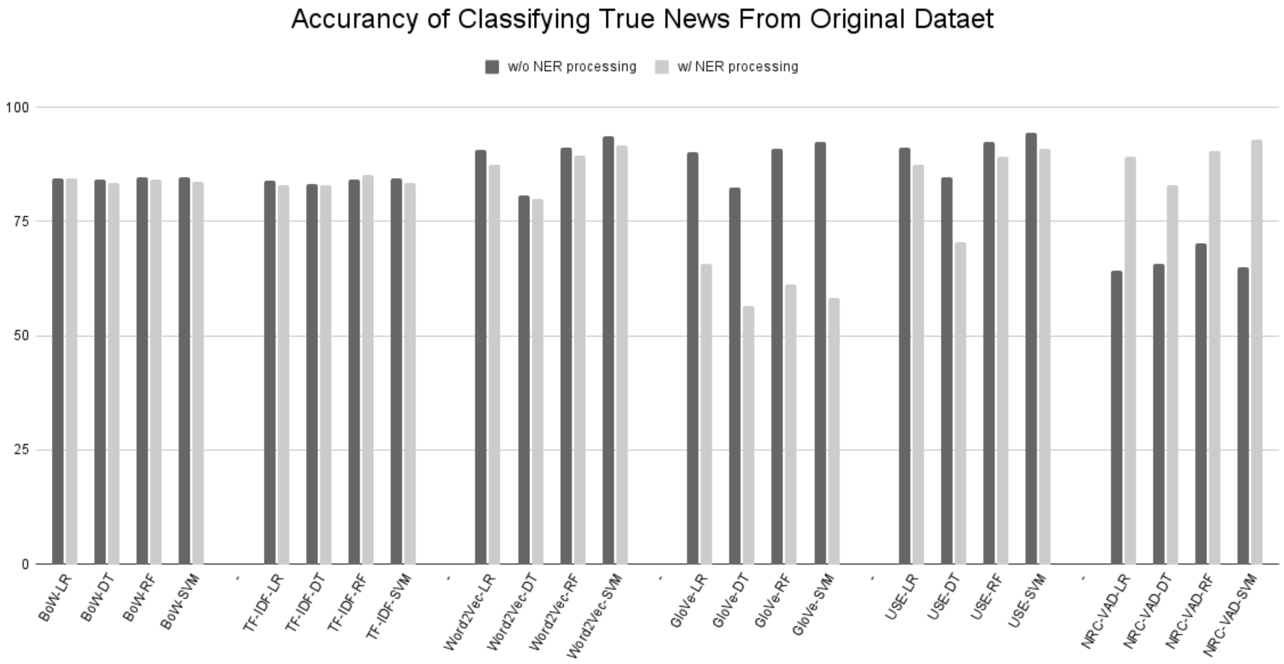

Figure 5: Average sentiment scores for fake and real news articles.

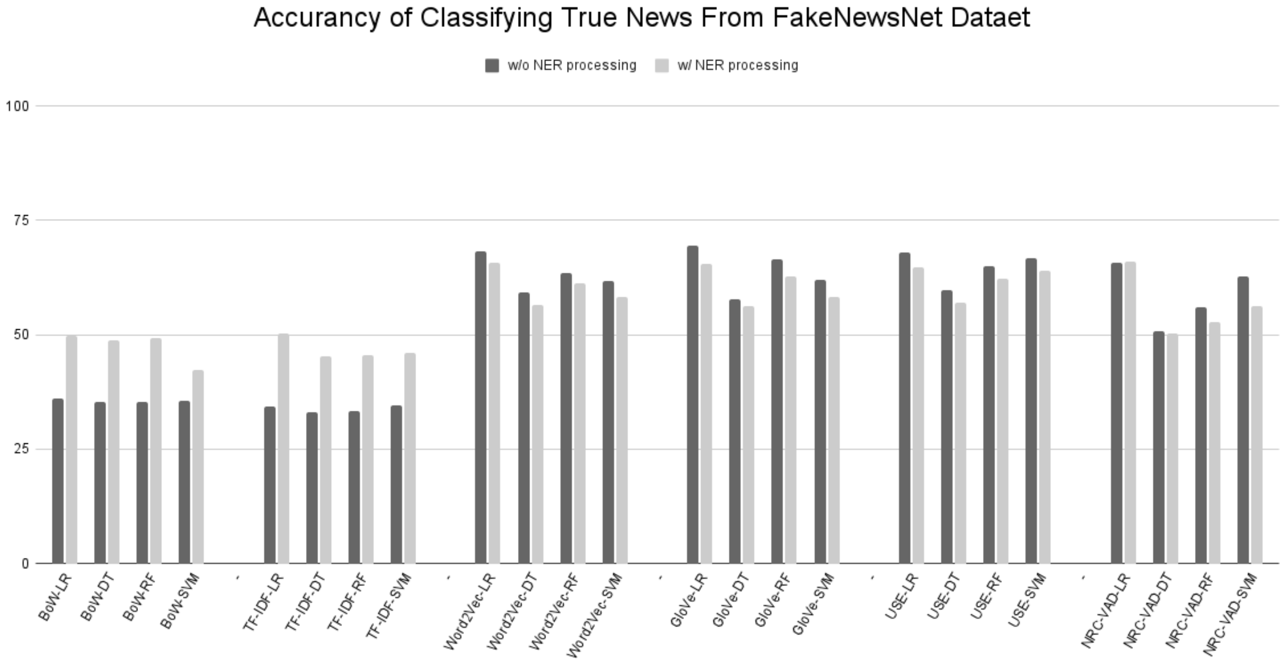

Figure 6: Boxplot of compound sentiment scores for fake and real news articles.

A EXPERIENMENT RESULTS

Table 5: Performance Metrics for Models without NER Preprocessing

Dataset	Feature	Model	Accuracy (%)	Precision (0/1)	Recall (0/1)
Original	Bag of Words	Logistic Regression	84.52	0.78 / 0.93	0.93 / 0.77
Original	Bag of Words	Decision Tree	84.25	0.78 / 0.93	0.93 / 0.76
Original	Bag of Words	Random Forest	84.61	0.78 / 0.93	0.85 / 0.84
Original	Bag of Words	Support Vector Machine	84.70	0.79 / 0.92	0.85 / 0.84
FakeNewsNet	Bag of Words	Logistic Regression	36.13	0.25 / 0.75	0.78 / 0.22
FakeNewsNet	Bag of Words	Decision Tree	35.43	0.25 / 0.76	0.80 / 0.21
FakeNewsNet	Bag of Words	Random Forest	35.46	0.25 / 0.76	0.80 / 0.21
FakeNewsNet	Bag of Words	Support Vector Machine	35.48	0.25 / 0.75	0.79 / 0.21
Original	TF-IDF	Logistic Regression	83.87	0.77 / 0.94	0.95 / 0.74
Original	TF-IDF	Decision Tree	83.25	0.77 / 0.92	0.84 / 0.82
Original	TF-IDF	Random Forest	84.07	0.77 / 0.94	0.94 / 0.75
Original	TF-IDF	Support Vector Machine	84.41	0.77 / 0.94	0.95 / 0.75
FakeNewsNet	TF-IDF	Logistic Regression	34.43	0.25 / 0.76	0.82 / 0.19
FakeNewsNet	TF-IDF	Decision Tree	33.09	0.25 / 0.74	0.82 / 0.17
FakeNewsNet	TF-IDF	Random Forest	33.26	0.25 / 0.74	0.82 / 0.17
FakeNewsNet	TF-IDF	Support Vector Machine	34.49	0.25 / 0.76	0.81 / 0.19
Original	Word2Vec	Logistic Regression	90.59	0.90 / 0.91	0.91 / 0.91
Original	Word2Vec	Decision Tree	80.74	0.81 / 0.80	0.77 / 0.84
Original	Word2Vec	Random Forest	91.18	0.89 / 0.93	0.93 / 0.90
Original	Word2Vec	Support Vector Machine	93.77	0.93 / 0.95	0.94 / 0.93
FakeNewsNet	Word2Vec	Logistic Regression	68.30	0.24 / 0.75	0.13 / 0.87
FakeNewsNet	Word2Vec	Decision Tree	59.27	0.23 / 0.74	0.27 / 0.70
FakeNewsNet	Word2Vec	Random Forest	63.56	0.22 / 0.74	0.18 / 0.78
FakeNewsNet	Word2Vec	Support Vector Machine	61.76	0.22 / 0.73	0.21 / 0.75
Original	GloVe	Logistic Regression	90.05	0.89 / 0.91	0.90 / 0.90
Original	GloVe	Decision Tree	82.35	0.83 / 0.82	0.79 / 0.85
Original	GloVe	Random Forest	90.90	0.89 / 0.92	0.92 / 0.90
Original	GloVe	Support Vector Machine	92.39	0.91 / 0.93	0.93 / 0.92
FakeNewsNet	GloVe	Logistic Regression	69.59	0.20 / 0.75	0.07 / 0.90
FakeNewsNet	GloVe	Decision Tree	57.74	0.25 / 0.75	0.36 / 0.65
FakeNewsNet	GloVe	Random Forest	66.55	0.22 / 0.75	0.14 / 0.84
FakeNewsNet	GloVe	Support Vector Machine	62.06	0.23 / 0.75	0.22 / 0.75
Original	USE	Logistic Regression	91.15	0.90 / 0.92	0.92 / 0.91
Original	USE	Decision Tree	84.65	0.86 / 0.84	0.81 / 0.88
Original	USE	Random Forest	92.31	0.92 / 0.93	0.92 / 0.92
Original	USE	Support Vector Machine	94.30	0.93 / 0.95	0.95 / 0.94
FakeNewsNet	USE	Logistic Regression	68.06	0.22 / 0.75	0.11 / 0.87
FakeNewsNet	USE	Decision Tree	59.69	0.24 / 0.75	0.28 / 0.70
FakeNewsNet	USE	Random Forest	64.93	0.21 / 0.74	0.15 / 0.81
FakeNewsNet	USE	Support Vector Machine	66.86	0.23 / 0.75	0.14 / 0.84
Original	VAD Score	Logistic Regression	64.35	0.63 / 0.65	0.59 / 0.69
Original	VAD Score	Decision Tree	65.73	0.65 / 0.66	0.60 / 0.71
Original	VAD Score	Random Forest	70.35	0.70 / 0.71	0.65 / 0.75
Original	VAD Score	Support Vector Machine	65.00	0.65 / 0.65	0.58 / 0.71
FakeNewsNet	VAD Score	Logistic Regression	65.79	0.26 / 0.75	0.20 / 0.81
FakeNewsNet	VAD Score	Decision Tree	50.91	0.25 / 0.75	0.48 / 0.52
FakeNewsNet	VAD Score	Random Forest	56.00	0.24 / 0.75	0.35 / 0.63
FakeNewsNet	VAD Score	Support Vector Machine	62.77	0.24 / 0.75	0.23 / 0.76

Table 6: Performance Metrics for Models with NER Preprocessing

Dataset	Feature	Model	Accuracy (%)	Precision (0/1)	Recall (0/1)
Original	Bag of Words	Logistic Regression	84.36	0.79 / 0.90	0.91 / 0.79
Original	Bag of Words	Decision Tree	83.45	0.78 / 0.90	0.84 / 0.83
Original	Bag of Words	Random Forest	84.16	0.79 / 0.90	0.90 / 0.79
Original	Bag of Words	Support Vector Machine	83.79	0.78 / 0.92	0.92 / 0.76
FNN	Bag of Words	Logistic Regression	49.89	0.22 / 0.73	0.39 / 0.53
FNN	Bag of Words	Decision Tree	48.82	0.21 / 0.72	0.40 / 0.52
FNN	Bag of Words	Random Forest	49.28	0.21 / 0.72	0.39 / 0.53
FNN	Bag of Words	Support Vector Machine	42.25	0.23 / 0.72	0.56 / 0.38
Original	TF-IDF	Logistic Regression	82.87	0.79 / 0.87	0.86 / 0.80
Original	TF-IDF	Decision Tree	83.06	0.81 / 0.86	0.85 / 0.81
Original	TF-IDF	Random Forest	85.24	0.82 / 0.88	0.88 / 0.83
Original	TF-IDF	Support Vector Machine	83.35	0.77 / 0.92	0.92 / 0.75
FNN	TF-IDF	Logistic Regression	50.29	0.21 / 0.72	0.35 / 0.55
FNN	TF-IDF	Decision Tree	45.33	0.23 / 0.73	0.52 / 0.43
FNN	TF-IDF	Random Forest	45.48	0.23 / 0.73	0.51 / 0.44
FNN	TF-IDF	Support Vector Machine	46.08	0.22 / 0.72	0.45 / 0.46
Original	Word2Vec	Logistic Regression	87.47	0.86 / 0.89	0.88 / 0.87
Original	Word2Vec	Decision Tree	79.87	0.80 / 0.79	0.76 / 0.83
Original	Word2Vec	Random Forest	89.53	0.87 / 0.92	0.92 / 0.87
Original	Word2Vec	Support Vector Machine	91.67	0.90 / 0.93	0.93 / 0.91
FNN	Word2Vec	Logistic Regression	65.63	0.29 / 0.76	0.26 / 0.79
FNN	Word2Vec	Decision Tree	56.51	0.25 / 0.75	0.38 / 0.63
FNN	Word2Vec	Random Forest	61.26	0.25 / 0.75	0.28 / 0.72
FNN	Word2Vec	Support Vector Machine	58.37	0.25 / 0.75	0.33 / 0.67
Original	GloVe	Logistic Regression	87.45	0.87 / 0.88	0.87 / 0.88
Original	GloVe	Decision Tree	79.36	0.80 / 0.79	0.75 / 0.83
Original	GloVe	Random Forest	89.15	0.87 / 0.91	0.90 / 0.88
Original	GloVe	Support Vector Machine	90.91	0.89 / 0.91	0.90 / 0.90
FNN	GloVe	Logistic Regression	65.54	0.26 / 0.75	0.20 / 0.80
FNN	GloVe	Decision Tree	56.39	0.25 / 0.75	0.38 / 0.63
FNN	GloVe	Random Forest	62.72	0.23 / 0.75	0.21 / 0.77
FNN	GloVe	Support Vector Machine	58.33	0.25 / 0.75	0.33 / 0.67
Original	USE	Logistic Regression	89.27	0.88 / 0.91	0.90 / 0.89
Original	USE	Decision Tree	82.90	0.83 / 0.83	0.80 / 0.85
Original	USE	Random Forest	90.54	0.88 / 0.93	0.92 / 0.89
Original	USE	Support Vector Machine	92.94	0.92 / 0.94	0.94 / 0.92
FNN	USE	Logistic Regression	64.69	0.25 / 0.75	0.21 / 0.79
FNN	USE	Decision Tree	56.95	0.23 / 0.74	0.32 / 0.65
FNN	USE	Random Forest	62.36	0.23 / 0.74	0.21 / 0.76
FNN	USE	Support Vector Machine	63.95	0.24 / 0.75	0.21 / 0.78
Original	VAD Score	Logistic Regression	63.71	0.62 / 0.65	0.59 / 0.68
Original	VAD Score	Decision Tree	65.48	0.65 / 0.66	0.60 / 0.70
Original	VAD Score	Random Forest	69.74	0.69 / 0.70	0.65 / 0.74
Original	VAD Score	Support Vector Machine	64.96	0.65 / 0.65	0.58 / 0.71
FakeNewsNet	VAD Score	Logistic Regression	66.10	0.25 / 0.75	0.18 / 0.82
FakeNewsNet	VAD Score	Decision Tree	50.25	0.24 / 0.75	0.48 / 0.51
FakeNewsNet	VAD Score	Random Forest	52.91	0.24 / 0.75	0.43 / 0.56
FakeNewsNet	VAD Score	Support Vector Machine	56.33	0.24 / 0.75	0.35 / 0.63