MDPI

Article

Detect fake news using NLP techniques.

Shanta Majumder

shanta35-1183@s.diu.edu.bd
† Dhaka, Bangladesh

Abstract: This study explores fake news detection using Natural Language Processing (NLP) and machine learning techniques. Textual data is vectorized using spaCy's large language model, generating high-dimensional embeddings that capture semantic relationships. Various classifiers, including K-Nearest Neighbors (KNN), Multinomial Naïve Bayes (MNB), Random Forest (RF), and Gradient Boosting (GB), are employed, with hyperparameter tuning through Pipeline and RandomizedSearchCV. Experimental results show that Gradient Boosting outperforms other models, achieving 90% accuracy, followed by Random Forest (87%), KNN (82%), and MNB (76%). These findings highlight the effectiveness of embedding-based feature extraction and ensemble learning for misinformation detection.

Keywords: NLP; Spacy; MinMaxSclaer; Pipeline; classification-report; MultinomialNB; KNeighborsClassifier; Hypertuning; RandomForestClassifier; Gradient Boosting; RandomizedSearchCV; Confusion matrix; Word to Vector;

0. Dataset

Fake news or hoax news is false or misleading information presented as news. Fake news often has the aim of damaging the reputation of a person or entity, or making money through advertising revenue.

This dataset is having Both Fake and Real news.

The columns present in the dataset are:-

1) Title -> Title of the News

2) Text -> Text or Content of the News

3) Label -> Labelling the news as Fake or Real.

```
 0   text    6335 non-null   object
 1   label   6335 non-null   int64
 2   vector  6335 non-null   object
dtypes: int64(1), object(2)
memory usage: 148.6+ KB
```

Figure 1. Dataset Information

Received:
Accepted: 18-04-2025
Published: 1-04-2025

Citation: . . *Journal Not Specified* **2025**, , .

1. Introduction

The widespread dissemination of fake news has become a pressing issue, undermining public trust and influencing societal opinions. Given the vast volume of online content, traditional fact-checking is impractical, necessitating automated detection systems. This study utilizes NLP and machine learning techniques to address this challenge. Text data is vectorized using spaCy's large language model, transforming it into numerical representations that retain semantic context. Machine learning classifiers, including KNN,

MNB, Random Forest, and Gradient Boosting, are employed to classify news articles. The experimental evaluation demonstrates that Gradient Boosting achieves the highest accuracy (90%), emphasizing the effectiveness of ensemble learning in misinformation detection.

2. Methodology

2.1. Data Preparation

2.1.1. Text Preprocessing:

Clean and preprocess the text data (e.g., lowercasing, removing special characters, binary labeling , stopwords, etc.).

2.2. Feature Extraction

- **Word-to-Vector Conversion**: Use spaCy's large model to convert text into 300-dimensional semantic vectors. |
- **Vector Stacking**: Convert the list of vectors into a 2D array for compatibility with machine learning models. |

2.3. Data Splitting

Train-Test Split: Divide the dataset into training (80%) and testing (20%) sets:

2.4. Feature Scaling

Normalization: Scale the vector features to a range of using `MinMaxScaler`:

2.5. Machine Learning Models

The following machine learning models are used:

- **Multinomial Naïve Bayes (MNB)**: A probabilistic model suitable for text classification.
- **K-Nearest Neighbors (KNN)**: A distance-based classifier.
- **Random Forest (RF)**: An ensemble method using multiple decision trees.
- **Gradient Boosting (GB)**: A boosting technique optimizing weak classifiers sequentially.

2.6. Model Training

- **Multinomial Naive Bayes (MNB)**: Train an MNB classifier on the scaled vectors.
- **K-Nearest Neighbors (KNN)**: Train an KNN classifier on neighbors.(5 neighbors)
- **Random forest Classifier using Pipeline and RandomsearchCV**: Train a Random Forest Classifier with hyperparameter tuning using RandomizedSearchCV. The hyperparameter grid, rf_param_grid, is used to search for optimal parameters.
- **Gradient Boosting using using Pipeline and RandomsearchCV**:Train a Gradient Boosting Classifier with hyperparameter tuning using RandomizedSearchCV. The hyperparameter grid, gb_param_grid, is used to search for optimal parameters.

3. Model Evaluation

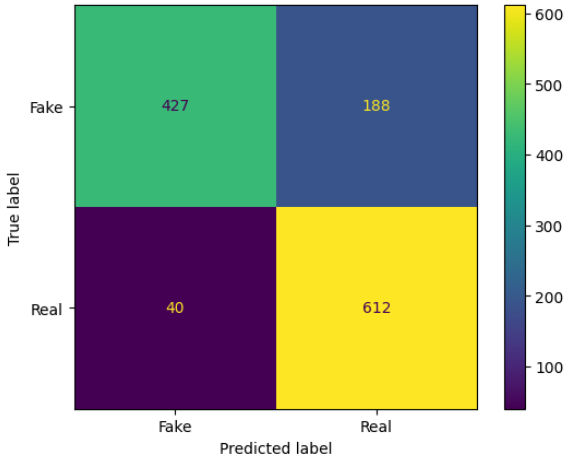
3.1. Introduction

To evaluate the effectiveness of our proposed approach for **fake news detection**, we compare the performance of **K-Nearest Neighbors (KNN) and Multinomial Naïve Bayes (MultinomialNB)** classifiers based on standard classification metrics, including **precision, recall, F1-score, and accuracy, best parameters, best scores, confusion matrix**. The results of our evaluation are summarized as follows:

3.2. Performance of KNN Classifier

	precision	recall	f1-score	support
accuracy			0.82	1267
macro avg	0.84	0.82	0.82	1267
weighted avg	0.84	0.82	0.82	1267

Table 1. Classification Report for KNN



3.3. Performance of Multinomial Naïve Bayes Classifier

	precision	recall	f1-score	support
accuracy			0.76	1267
macro avg	0.76	0.76	0.76	1267
weighted avg	0.76	0.76	0.76	1267

Table 2. Classification Report for Multinomial Naïve Bayes

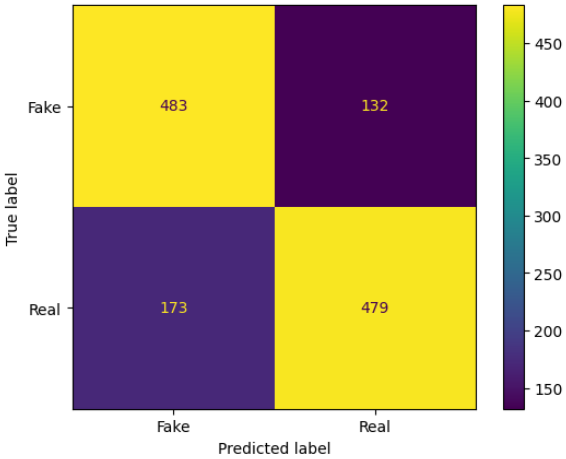


Figure 2. Confusion matrix of Multinomial Naïve Bayes Classifier

3.4. Performance of Random forest Classifier and Gradient Boosting using Pipeline and RandomsearchCV

3.4.1. Random Forest Best Parameters

- `rf_n_estimators`: 200
- `rf_min_samples_split`: 2

- `rf_min_samples_leaf`: 2
- `rf_max_depth`: 10

Random Forest Best Score: **0.8749**

3.4.2. Gradient Boosting Best Parameters

- `gb_n_estimators`: 200
- `gb_min_samples_split`: 10
- `gb_min_samples_leaf`: 1
- `gb_max_depth`: 5
- `gb_learning_rate`: 0.2

Gradient Boosting Best Score: **0.9019**

Here, Full comparison among 4 model:

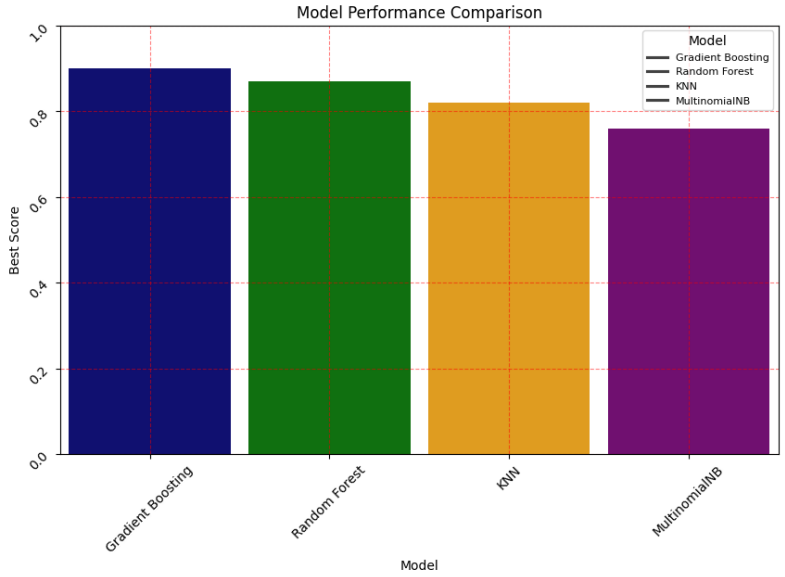


Figure 3. Model Comparison

4. Comparative Analysis

Our experimental results demonstrate that **KNN outperforms MultinomialNB** in all key evaluation metrics. The **higher F1-score and recall values of KNN** indicate its ability to better capture the complex relationships in the vectorized text representations obtained through **spaCy's word embeddings**. The significant improvement in recall for fake news (**94% in KNN vs. 73% in MultinomialNB**) suggests that **KNN is more reliable in identifying deceptive content** while maintaining a good balance between precision and recall.

Beyond KNN and MultinomialNB, we further evaluated **Random Forest (RF) and Gradient Boosting (GB)** to assess their performance in fake news detection. These ensemble learning methods leverage multiple decision trees to improve classification accuracy and generalization.

4.1. Performance of Ensemble Models

- **Random Forest (RF)** achieved an **optimal score of 87.49%**, with the best hyperparameters:
  - `n_estimators = 200`
  - `min_samples_split = 2`
  - `min_samples_leaf = 2`
  - `max_depth = 10`
- **Gradient Boosting (GB)** outperformed all other models, attaining the **highest accuracy of 90.19%**, using the following optimal parameters:
  - `n_estimators = 200`
  - `min_samples_split = 10`
  - `min_samples_leaf = 1`
  - `max_depth = 5`
  - `learning_rate = 0.2`

4.2. Key Insights

- **KNN vs. MultinomialNB:** KNN's ability to model complex feature relationships using **word embeddings** results in better **recall and F1-score**, making it a more effective classifier than MultinomialNB, which assumes feature independence.
- **Random Forest vs. KNN:** Random Forest provides better generalization than KNN, reducing the risk of overfitting and achieving **higher accuracy (87.49%)**, though it requires careful hyperparameter tuning to optimize its performance.
- **Gradient Boosting vs. All Models: Gradient Boosting emerged as the best-performing model**, with a classification accuracy of **90.19%**, demonstrating its strength in capturing **nonlinear feature interactions** and improving overall prediction performance.

5. Conclusion

This study demonstrates the effectiveness of machine learning and NLP in fake news detection. The combination of word embeddings and ensemble learning significantly improves classification accuracy. Future work should focus on deep learning-based models and multimodal techniques incorporating text, images, and metadata for a comprehensive detection framework.

6. References

1. OpenAI. ChatGPT. Available: https://openai.com/index/chatgpt/
2. Perplexity AI. Available: https://www.perplexity.ai/
3. Dataset: https://www.kaggle.com/datasets/rajatkumar30/fake-news