Sports vs Politics Text Classification using Machine Learning

Om Chandra Sharma
Roll No: B23CS1048

1 Introduction

Text classification is a fundamental problem in Natural Language Understanding (NLU), where textual data is categorized into predefined classes. In this project, we design and evaluate a binary text classification system that classifies sentences into two domains: **Sports** and **Politics**.

The objective is to compare multiple machine learning techniques using different feature representations and evaluate their performance quantitatively.

2 Dataset Preparation

2.1 Data Collection

The dataset was constructed using an automated Wikipedia crawling pipeline implemented in Python.

Domain-specific keyword sets were automatically generated from the following category pages:

- `https://en.wikipedia.org/wiki/Category:Sports`
- `https://en.wikipedia.org/wiki/Category:Politics`

Up to 500 keywords per domain were extracted from article titles listed under these categories.

2.2 Title-Based Filtering

Only Wikipedia pages whose titles contained at least one domain-specific keyword were included. The following filters were applied:

- Disambiguation pages were excluded.
- Meta pages (titles containing ":") were excluded.
- Only domain-relevant titles were processed.

This strict filtering prevented topic drift and ensured domain consistency.

2.3 Sentence Extraction

For each valid page:

- Paragraph text was extracted.
- Text was tokenized into sentences using NLTK.
- Sentences with fewer than 8 words were discarded.

The process continued until 25,000 sentences were collected per class.

2.4 Final Dataset Statistics

- Total samples: 50,000
- Sports sentences: 25,000
- Politics sentences: 25,000
- Balanced dataset

Each sample contains:

- A sentence
- A binary label (0 = Sports, 1 = Politics)

3 Exploratory Data Analysis

Sports sentences frequently contained domain-specific terms such as:
match, tournament, player, goal, league, championship, athlete, coach
Politics sentences frequently contained:
government, election, parliament, policy, minister, democracy, senate, legislation
The balanced dataset ensures unbiased model training.

4 Feature Representation

Three feature representation techniques were evaluated:

4.1 Bag of Words (BoW)

BoW represents text as frequency counts of words. Word order is ignored, and only occurrence frequency is considered.

4.2 TF-IDF

TF-IDF (Term Frequency–Inverse Document Frequency) assigns weights to words based on their importance across documents, reducing the impact of common words.

4.3 N-grams

N-grams capture sequences of words. In this project, unigram and bigram features were used to incorporate contextual information.

5 Machine Learning Models

Three classifiers were implemented:

5.1 Naive Bayes

A probabilistic classifier based on Bayes' theorem with independence assumptions between features.

5.2 Logistic Regression

A linear model that estimates class probabilities using a logistic function.

5.3 Support Vector Machine (SVM)

A margin-based classifier that finds the optimal hyperplane separating the two classes.

6 Experimental Setup

The dataset was split into:

- 80% Training data
- 20% Testing data

Performance metrics used:

- Accuracy
- Precision
- Recall
- F1-Score

7 Results and Analysis

7.1 Feature Representation Comparison

To evaluate the impact of different feature extraction techniques, three representations were compared using the Naive Bayes classifier:

- Bag of Words
- TF-IDF

- N-grams (Unigram + Bigram)

The performance metrics are summarized below:

Feature Method	Accuracy	Precision	Recall	F1-score
Bag of Words	0.9639	0.9579	0.9700	0.9639
TF-IDF	0.9585	0.9522	0.9650	0.9585
N-grams (1,2)	**0.9659**	**0.9597**	**0.9722**	**0.9659**

Table 1: Feature Representation Comparison (Using Naive Bayes)

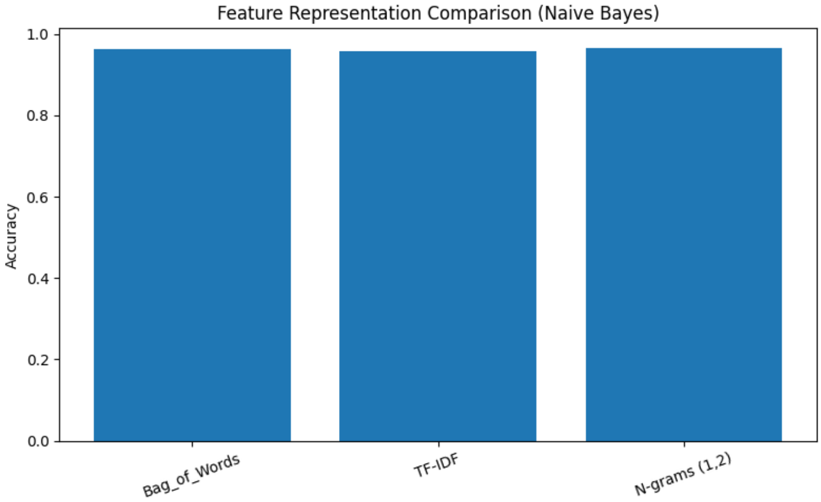

Figure 1: Accuracy Comparison of Feature Representations

Observation:

The N-gram representation achieved the highest performance across all metrics. The inclusion of bigrams allowed the model to capture contextual patterns such as:

- ”prime minister”
- ”world cup”
- ”public policy”
- ”national team”

This contextual information improves discriminative power compared to single-word features.

7.2 Model Comparison Using Best Feature (N-grams)

Since N-grams (1,2) achieved the highest accuracy, it was selected as the feature representation for model comparison.

Four classifiers were evaluated:

- Naive Bayes
- Logistic Regression
- Linear SVM
- Random Forest

Model	Accuracy	Precision	Recall	F1-score
Naive Bayes	**0.9659**	0.9597	**0.9722**	**0.9659**
Logistic Regression	0.9638	0.9575	0.9702	0.9638
Linear SVM	0.9617	0.9568	0.9666	0.9617
Random Forest	0.9444	0.9233	0.9686	0.9454

Table 2: Model Comparison Using N-gram Features

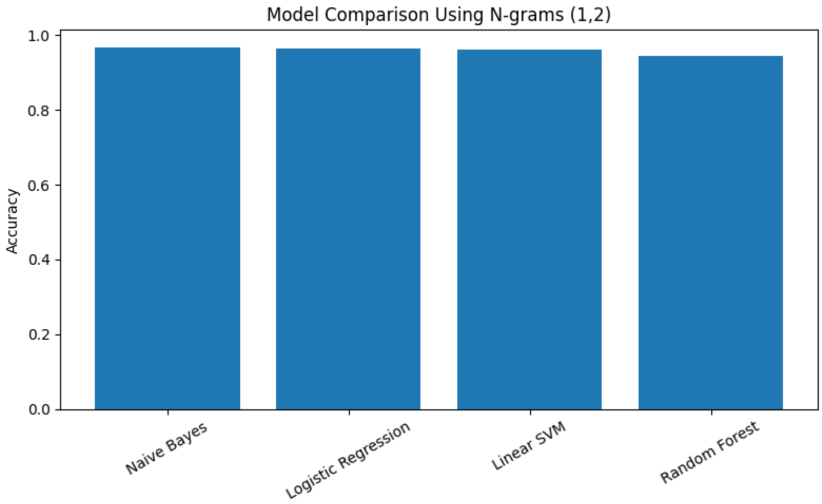

Figure 2: Accuracy Comparison of Machine Learning Models

7.3 Confusion Matrix Analysis

Confusion matrices were generated for each classifier to analyze misclassification patterns.

- Naive Bayes showed the most balanced classification.
- Logistic Regression performed similarly with slightly lower recall.
- SVM performed competitively but slightly below Naive Bayes.
- Random Forest showed lower precision, indicating more false positives.

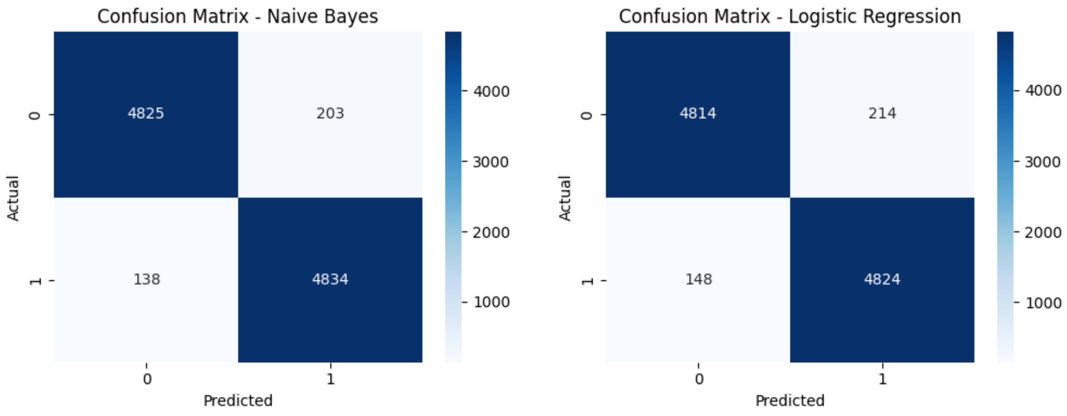

Figure 3: Confusion Matrices (Naive Bayes and Logistic Regression)

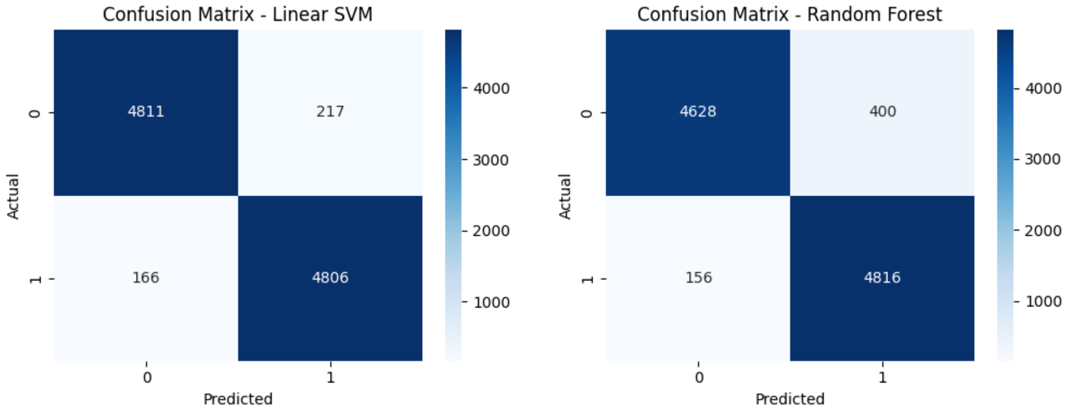

Figure 4: Confusion Matrices (Linear SVM and Random Forest)

7.4 Discussion

Several important observations can be made:

1. N-grams significantly improved classification accuracy by capturing contextual word relationships.
2. Naive Bayes performed surprisingly well despite its strong independence assumptions.
3. Logistic Regression and SVM performed comparably due to their effectiveness in high-dimensional spaces.
4. Random Forest underperformed compared to linear models, likely because text features are sparse and high-dimensional, which tree-based models handle less efficiently.

7.5 Overall Best Configuration

The best performing configuration was:

- Feature Representation: N-grams (Unigram + Bigram)
- Model: Naive Bayes
- Accuracy: 96.59%

This demonstrates that classical probabilistic models combined with contextual feature representations are highly effective for domain-based text classification tasks.

8 Limitations

Although the proposed system achieves high classification accuracy, several limitations exist:

8.1 Source Bias

The dataset was collected exclusively from Wikipedia. While Wikipedia provides structured and reliable content, it may not reflect the writing style of real-world news articles, social media posts, or informal text. Therefore, the model may not generalize well to other domains such as tweets or live news feeds.

8.2 Title-Based Filtering Assumption

The data collection process relied on strict title-based keyword filtering. Although this ensures domain consistency, it may exclude relevant articles whose titles do not explicitly contain domain keywords. Conversely, some accepted articles may still contain mixed-domain information.

8.3 Sentence-Level Classification

The system performs classification at the sentence level. Some sentences may lack sufficient context to clearly indicate whether they belong to sports or politics. In real-world scenarios, document-level classification may provide more reliable predictions.

8.4 Limited Feature Representations

The project evaluates traditional feature representations such as Bag of Words, TF-IDF, and N-grams. These approaches ignore deeper semantic relationships and word embeddings. Modern contextual embedding models (e.g., BERT, RoBERTa) could potentially improve performance.

8.5 Model Simplicity

Only classical machine learning models were evaluated. While Naive Bayes and Logistic Regression performed well, deep learning architectures were not explored due to computational constraints.

8.6 Binary Classification Scope

The system is limited to binary classification (Sports vs Politics). Real-world applications may require multi-class classification involving additional domains such as Business, Technology, Health, and Entertainment.

8.7 Language Restriction

The dataset contains only English text. The model cannot generalize to multilingual or code-mixed data without retraining.

9 Conclusion

This project successfully implemented a binary text classification system distinguishing Sports and Politics sentences. A large, balanced dataset of 50,000 samples was constructed using automated domain-based crawling. Multiple feature representations and machine learning techniques were evaluated and compared quantitatively. The results demonstrate the effectiveness of classical ML techniques in domain-based text classification.