Sports vs Politics Text Classification

Project Report

Submitted by

Meet Tilala

B23CS1036

January 2026

Contents

Abstract

Text classification is a fundamental task in Natural Language Processing (NLP) with applications ranging from information retrieval to automated content moderation. This project presents a supervised machine learning system for binary text classification, categorizing sentences into two domains: *Sports* and *Politics*.

A large-scale, fully balanced dataset consisting of 50,000 sentences was automatically constructed using Wikipedia as a data source. Domain-specific keyword extraction, strict title-based filtering, and sentence-level preprocessing were employed to ensure high domain purity and reproducibility. Multiple feature representation techniques, including Bag of Words (BoW), TF-IDF, and N-grams, were evaluated alongside four machine learning classifiers: Naive Bayes, Logistic Regression, Linear Support Vector Machine (SVM), and Random Forest.

1 Introduction

With the exponential growth of digital text data, automatic document classification has become increasingly important. News articles, social media posts, blogs, and online encyclopedias often span multiple domains, making manual categorization impractical at scale. Automated text classification systems address this challenge by leveraging machine learning techniques to assign semantic labels to text.

This project focuses on binary domain classification between **Sports** and **Politics**. These domains were chosen due to their high lexical overlap (e.g., terms such as *team*, *campaign*, *match*, *election*) and distinct contextual patterns, making them suitable for evaluating feature extraction and model effectiveness.

The project emphasizes:

- Large-scale automated dataset construction
- Comparative analysis of feature representations
- Evaluation of multiple supervised learning models
- Quantitative performance comparison
- Deployment for real-time inference

Project Repository: The complete source code, dataset generation scripts, trained models, and execution instructions are publicly available on GitHub:

https://github.com/Meet-Tilala/Sports_politics_classifier.git

2 Problem Statement

The objective of this project is to design and implement a supervised text classification system that:

1. Accepts a raw text sentence as input
2. Extracts meaningful numerical features from the text
3. Applies a trained machine learning model
4. Classifies the text into one of two categories:
    - Sports
    - Politics

The system must be evaluated using standard classification metrics and the best-performing configuration must be deployed for real-time usage.

3 Dataset Construction

3.1 Data Source

Wikipedia was selected as the data source due to its:

- Rich, well-structured content
- Clear domain categorization
- Public accessibility
- Reproducibility

The following category pages were used as entry points:

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

3.2 Automated Keyword Generation

Instead of manually selecting domain-specific keywords, an automated approach was adopted:

- Article titles from category pages were extracted

- Up to 500 keywords were generated per domain
- Keywords directly reflected Wikipedia's taxonomy

This approach minimized human bias and improved reproducibility.

3.3 Title-Based Filtering

To ensure domain consistency and prevent topic drift, strict filtering rules were applied:

- Article titles must contain at least one domain keyword
- Disambiguation pages were excluded
- Meta pages (titles containing ":") were excluded

3.4 Sentence Extraction

For each valid article:

1. Paragraph text was extracted
2. Text was tokenized into sentences using NLTK
3. Sentences shorter than 8 words were discarded

3.5 Final Dataset Statistics

Table 1: Final Dataset Composition

Category	**Sentences**	**Label**
Sports	50,000	0
Politics	50,000	1
Total	100,000	–

The dataset is fully balanced, eliminating class imbalance issues.

4 Feature Representation Techniques

4.1 Bag of Words (BoW)

The Bag of Words model represents text as a vector of word frequency counts. While simple and computationally efficient, it ignores word order and semantic context.

4.2 TF-IDF

TF-IDF weighs words based on their importance:

$$\text{TF-IDF}(t, d) = \text{TF}(t, d) \times \log\left(\frac{N}{DF(t)}\right)$$

This reduces the influence of common words and emphasizes discriminative terms.

4.3 N-grams (Unigram + Bigram)

N-grams capture local context by considering word sequences. Examples include:

- "Championship"
- "president"
- "peace treaty"

This representation proved effective for domain distinction.

5 Machine Learning Models

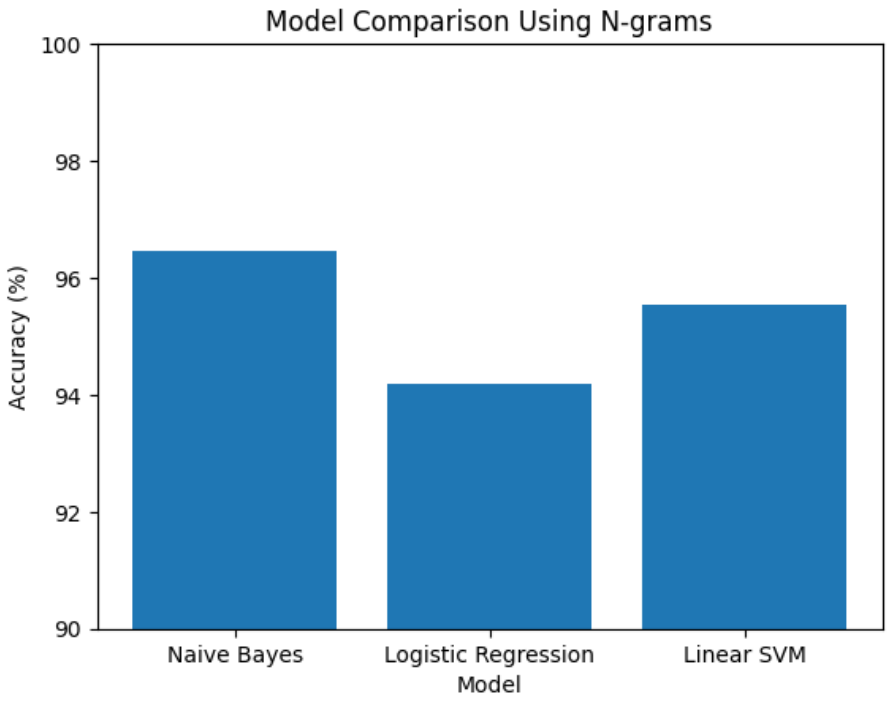
The following supervised classifiers were evaluated:

- Naive Bayes
- Logistic Regression
- Linear Support Vector Machine (SVM)
- Random Forest

Each model was trained and tested using identical data splits to ensure fair comparison.

6 Experimental Setup

- Training set: 75%
- Testing set: 25%
- Evaluation metrics:
  - Accuracy
  - Precision
  - Recall
  - F1-score



7 Results and Analysis

7.1 Feature Comparison Using Naive Bayes

Table 2: Feature Representation Comparison

Feature	**Accuracy**
Bag of Words	94.19%
TF-IDF	96.35%
N-grams (1,2)	**96.47%**

7.2 Model Comparison Using N-grams

Table 3: Classifier Performance Comparison

Model	**Accuracy**
Naive Bayes	**96.47%**
Logistic Regression	94.19%
Linear SVM	95.55%

7.3 Best Configuration

- Feature: N-grams (1,2)

- Model: Naive Bayes
- Accuracy: 96.47%

8 Model Deployment

The best-performing model was serialized using `joblib` and integrated into a real-time classification pipeline. Users can input any sentence, and the system outputs:

- **SPORTS**
- **POLITICS**

9 Conclusion

This project demonstrates the effectiveness of traditional machine learning techniques for domain-specific text classification. Automated dataset construction from Wikipedia ensured scalability and reproducibility. Experimental results confirm that contextual features such as N-grams significantly enhance classification performance. The deployed model enables accurate real-time predictions and can be extended to multi-class classification or deep learning-based approaches in future work.

10 Future Work

- Extension to multi-class classification
- Use of word embeddings (Word2Vec, GloVe)
- Transformer-based models (BERT)
- Cross-domain generalization analysis