# PROBLEM 4: SPORTS OR POLITICS

Jahnavi Gajera (M25CSA012)

February 13, 2026

# Introduction

The objective of this project is to design and evaluate a machine learning classifier capable of distinguishing between text documents categorized as "Sport" or "Politics." Text classification is a fundamental task in Natural Language Processing (NLP) with applications ranging from automated news aggregation to sentiment analysis.

# Data Collection and Description

## Data Sourcing

For this task, I used the **20 Newsgroups dataset**, a standard benchmark in the ML community. Specifically, i extracted four sub-categories to create a binary-like classification environment:

- **Sports:** *rec.sport.baseball* and *rec.sport.hockey*.
- **Politics:** *talk.politics.mideast* and *talk.politics.guns*.

## Dataset Analysis

The dataset consists of approximately 3,000 documents. Initial analysis showed that the "Sports" category frequently contains terms related to physical action and statistics (e.g., "puck," "homerun," "goalie"), while "Politics" text is dense with legal and geographic terminology (e.g., "legislation," "Israel," "amendment").

# Feature Representation Techniques

To convert raw text into a format understandable by algorithms, three techniques were evaluated:

## Bag-of-Words (BoW)

This model represents text by counting how many times each word appears in a document. While it is simple and easy to implement, it treats all words equally. As a result, very common words (such as "the", "is", "and") may receive high importance even though they do not contribute much meaningful information for classification. This can reduce the overall effectiveness of the model.

## N-gram Representation

It (e.g., bi-grams or tri-grams) considers sequences of words instead of individual words. This helps capture some contextual information and word order. However, using n-grams significantly increases the number of features, leading to:

- Higher memory usage
- Increased computational cost

- Greater risk of overfitting, especially with limited data

Therefore, although n-grams can improve context understanding, they make the model more complex and less efficient.

### TF-IDF Representation (Selected Approach)

TF-IDF (Term Frequency–Inverse Document Frequency) was selected because it provides a balanced and effective representation of text. Unlike Bag-of-Words, TF-IDF reduces the weight of very common words and gives higher importance to words that are more distinctive within a document.

Compared to n-grams, TF-IDF keeps the feature space more manageable while still highlighting the most informative terms. This results in:

- Better discrimination between document categories
- Improved computational efficiency
- Reduced risk of overfitting

### Final Justification

Overall, TF-IDF was chosen because it improves classification performance by focusing on meaningful words while maintaining lower complexity compared to n-gram models. It offers a good trade-off between accuracy, efficiency, and scalability, making it more suitable for this project.

## Machine Learning Techniques

We compared three distinct supervised learning algorithms:

1. **Multinomial Naive Bayes:**It is a probabilistic learner based on Bayes' Theorem. It is particularly effective for text due to its efficiency with high-dimensional sparse data.
2. **Support Vector Machines (SVM):** It aims to find the hyperplane in a high-dimensional space that maximizes the margin between the two classes.
3. **Logistic Regression:** It is a linear model that estimates the probability of a class label using the logistic function.

# Quantitative Comparison and Results

The following tables summarize the performance of the three models based on the experimental results.

# Detailed Classification Results

## Naive Bayes Results

**Overall Accuracy: 0.88**

| Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| rec.sport.baseball | 0.81 | 0.92 | 0.86 | 198 |
| rec.sport.hockey | 0.92 | 0.88 | 0.90 | 209 |
| talk.politics.guns | 0.87 | 0.87 | 0.87 | 178 |
| talk.politics.mideast | 0.93 | 0.83 | 0.88 | 184 |
| **Accuracy** | | 0.88 (769 samples) | | |
| Macro Avg | 0.88 | 0.87 | 0.88 | 769 |
| Weighted Avg | 0.88 | 0.88 | 0.88 | 769 |

Table 1: Classification Report for Naive Bayes

## Support Vector Machine (SVM) Results

**Overall Accuracy: 0.86**

| Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| rec.sport.baseball | 0.77 | 0.94 | 0.85 | 198 |
| rec.sport.hockey | 0.90 | 0.85 | 0.87 | 209 |
| talk.politics.guns | 0.87 | 0.85 | 0.86 | 178 |
| talk.politics.mideast | 0.94 | 0.80 | 0.87 | 184 |
| **Accuracy** | | 0.86 (769 samples) | | |
| Macro Avg | 0.87 | 0.86 | 0.86 | 769 |
| Weighted Avg | 0.87 | 0.86 | 0.86 | 769 |

Table 2: Classification Report for Support Vector Machine

### Logistic Regression Results

**Overall Accuracy: 0.88**

| Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| rec.sport.baseball | 0.78 | 0.93 | 0.85 | 198 |
| rec.sport.hockey | 0.92 | 0.87 | 0.89 | 209 |
| talk.politics.guns | 0.89 | 0.85 | 0.87 | 178 |
| talk.politics.mideast | 0.95 | 0.84 | 0.89 | 184 |
| **Accuracy** | | 0.88 (769 samples) | | |
| Macro Avg | 0.88 | 0.87 | 0.88 | 769 |
| Weighted Avg | 0.88 | 0.88 | 0.88 | 769 |

Table 3: Classification Report for Logistic Regression

### Summary of Accuracy

| Model | Overall Accuracy |
|---|---|
| Naive Bayes: | 0.88 |
| SVM (Linear Kernel) | 0.86 |
| Logistic Regression | 0.88 |

Table 4: Comparative Accuracy of the Three ML techniques.

**Naive Bayes Results:**

- Weighted Avg F1-Score: 0.88
- Observation: Highest recall for Baseball (0.92).

**Logistic Regression Results:**

- Weighted Avg F1-Score: 0.88
- Observation: Exceptional precision for Middle East Politics (0.95).

## System Limitations

Despite high accuracy, the system faces several challenges:

- **Sarcasm:** It cannot detect when a political term is used sarcastically in a sports context or when a sports term is used in a political context.
- **Overlapping Vocabulary:** Terms like "campaign", "club", and "win" appear in both domains, leading to misclassification.
- **Short Text:** It cannot work well on very short documents (less than 10 words) and makes more mistakes with them.

# Conclusion

Naive Bayes and Logistic Regression correctly predicted the category 88% of the time, which is slightly better than SVM, which had an accuracy of 86% on this dataset. In simple terms, this means Naive Bayes and Logistic Regression made fewer mistakes overall compared to SVM.

For real-world use, Logistic Regression would be a better choice. This is because it is especially good at correctly identifying political content, meaning it is more precise and less likely to wrongly label non-political content as political. This makes it more reliable when accurate political classification is important