REPORT: Sports vs Politics News Classification

Name: Avani Rai
Roll No.: B22CS094

---

1. Introduction

In this project, the objective is to build a machine learning classifier that can read a news document and classify it into one of two categories:

- **Sports News**
- **Politics News**

The overall pipeline of this project includes:

1. Dataset collection
2. Data preprocessing
3. Feature extraction using NLP techniques like Bag of Words, TF-IDF, and n-grams
4. Training and evaluation of multiple ML classifiers
5. Quantitative comparison of models
6. Discussion of limitations and future improvements

---

2. Dataset Collection

2.1 Data Source

The dataset used in this project was collected from Kaggle. Specifically, the dataset is based on the **BBC News Classification Dataset**, which contains news articles labeled into five categories:

- Business
- Entertainment
- Politics
- Sport
- Technology

Dataset link (Kaggle):
https://www.kaggle.com/datasets

The dataset was downloaded as a ZIP file, extracted, and loaded into Google Colab for further processing.

2.2 Motivation for Choosing BBC Dataset

The BBC News dataset was chosen because:

- It contains clean and well-written news articles
- Labels are already provided
- It includes both “sport” and “politics” categories required for this task

---

3. Dataset Description and Analysis

3.1 Dataset Structure

The dataset file `bbc_data.csv` contains two important columns:

- **data**: the actual news article text

- **labels**: the category label of the article

Example:

data	label
Arsenal won the match…	sport
Prime Minister announced…	politics

3.2 Dataset Size

The dataset contains 2225 total articles across five categories. The distribution is:

- Sport: 511 articles
- Politics: 417 articles
- Business: 510 articles
- Tech: 401 articles
- Entertainment: 386 articles

Since this project focuses only on binary classification, only the sport and politics articles were selected.

Final dataset used:

- Sport: 511
- Politics: 417
- Total: 928 articles

3.3 Data Preprocessing

The preprocessing steps included:

- Filtering only sport and politics documents
- Splitting dataset into training and testing sets
- Applying feature extraction methods

3.4 Train-Test Split

The dataset was split using an 80-20 ratio:

- Training set: 80%
- Testing set: 20%

Since the data was in order , like all the sports news followed by politics news, I took a random seed so that it can be split properly.

---

4. Feature Representation Techniques

Machine learning models cannot directly work with raw text. Therefore, text must be converted into numerical feature vectors.

Three feature extraction methods could be used as suggested in the question:

4.1 Bag of Words (BoW)

Bag of Words represents each document by the frequency of words appearing in it.

- Simple and effective
- Ignores grammar and word order
- Vocabulary size can become large

Implementation:

```
CountVectorizer()
```

4.2 TF-IDF (Term Frequency-Inverse Document Frequency)

TF-IDF improves BoW by down-weighting common words and emphasizing rare but informative words.

Advantages:

- Reduces impact of stopwords
- Better representation than raw frequency

Implementation:

```
TfidfVectorizer()
```

4.3 N-Grams

N-grams include sequences of words instead of only single tokens.

Example:

- Unigram: “prime”
- Bigram: “prime minister”

This captures more context than BoW.

Implementation:

```
TfidfVectorizer(ngram_range=(1,2))
```

---

5. Machine Learning Models Compared

At least three ML classifiers were trained and compared:

5.1 Multinomial Naive Bayes

Naive Bayes is a probabilistic classifier based on Bayes' theorem.

Strengths:

- Fast
- Works well with word frequencies
- Strong baseline for text classification

Weakness:

- Assumes word independence

5.2 Logistic Regression

Logistic Regression is a linear classifier that models the probability of a class.

Strengths:

- Strong performance on text classification
- Works well with TF-IDF features

5.3 Support Vector Machine (SVM)

SVM is one of the best models for high-dimensional sparse text data.

Strengths:

- Very high accuracy
- Effective for document classification

Weakness:

- Computationally heavier than Naive Bayes

. Quantitative Results and Comparison

After preprocessing the dataset and selecting only the *Sport* and *Politics* categories, the documents were transformed into numerical feature vectors using **TF-IDF with bigrams**. The following feature extraction method was applied:

```
from sklearn.feature_extraction.text import TfidfVectorizer


tfidf = TfidfVectorizer(stop_words="english", ngram_range=(1,2))


X_train_vec = tfidf.fit_transform(X_train)

X_test_vec = tfidf.transform(X_test)
```

This representation captures both individual important words (unigrams) and meaningful word pairs (bigrams), such as *“prime minister”* or *“world cup”*, which are strong indicators of political or sports-related content.

Accuracy Results

Model	Accuracy

Naive Bayes	**0.9946**
Logistic Regression	**0.9839**
Support Vector Machine (SVM)	**0.9946**

Observations

- **Naive Bayes achieved the highest accuracy (99.46%)**, showing that probabilistic models perform extremely well on topic-based classification tasks where word frequency is highly informative.
- **SVM also achieved the same highest accuracy (99.46%)**, indicating that the dataset is linearly separable with strong feature representation.
- **Logistic Regression performed slightly lower (98.39%)**, but still achieved very strong classification performance.

Overall, all three models performed exceptionally well due to the clean nature of the BBC news dataset and the effectiveness of TF-IDF bigram features.

---

7. Limitations of the System

Despite strong performance, the system has limitations:

1. Limited Categories

Only sport and politics were used. Real-world news includes many overlapping topics.

2. No Deep Semantic Understanding

Traditional ML models rely on word statistics, not meaning.

3. Dataset Bias

BBC articles are formal and clean. Performance may drop on social media or noisy text.

4. No Contextual Embeddings

Modern NLP models like BERT could achieve better generalization.

5. Binary Restriction

The classifier cannot classify other categories like business or entertainment.

Conclusion

> “Using TF-IDF with bigram features, Naive Bayes and SVM achieved the highest accuracy of 99.46%, while Logistic Regression achieved 98.39%. These results demonstrate that classical machine learning methods are highly effective for binary news topic classification.”