LANGUAGE DETECTION SYSTEM USING MACHINE LEARNING

REPORT

ITSOLERA
PVT LTD

MACHINE LEARNING (INTERNS)

TEAM ETA

Zarnab Zafar (Team Lead)
Muhammad Mustafa Shah
Ayesha Majeed
Wajahat Hussain
Dil Nawaz
Muhammad Abu Bakar
Ali Khan

INTRODUCTION:

1. BACKGROUND:

Language detection is a key task within the field of natural language processing (NLP), which involves identifying the language of a given text. This task is critical for many applications, such as machine translation, content analysis, and search engines. Traditionally, rule-based methods have been used for language detection, but they often face limitations when dealing with complex language structures, mixed content, and shorter texts.

To overcome these limitations, machine learning models such as **Naive Bayes** and **Support Vector Machines (SVM)** have emerged as powerful alternatives. These models can learn from large datasets and accurately identify language patterns, even in noisy or ambiguous text samples.

2. PROBLEM STATEMENT:

With the growing importance of multilingual content, the ability to accurately detect the language of a given text is becoming essential. Traditional language detection approaches often fail when dealing with short texts or mixed languages, which can negatively impact downstream applications like translation or content categorization. Therefore, a more robust, machine learning-driven solution is needed to improve accuracy and handle diverse language data effectively.

3. OBJECTIVES:

1. Develop a machine learning-based system capable of detecting the language of a text with high accuracy.
2. Evaluate the performance of **Naive Bayes** and **SVM** models for language detection, identifying their strengths and weaknesses.
3. Provide recommendations on feature engineering and model optimization for enhanced performance.
4. Assess the system's applicability in real-world scenarios, such as automatic language detection in translation services.

4. DATA ANALYSIS AND PREPARATION:

Dataset Description

The dataset used in this project consists of text samples from multiple languages, including English, French, Spanish, Portuguese, Italian, Russian, Dutch, German, Arabic, and several others. The dataset is labeled, meaning each text sample is tagged with its respective language, allowing for supervised learning.

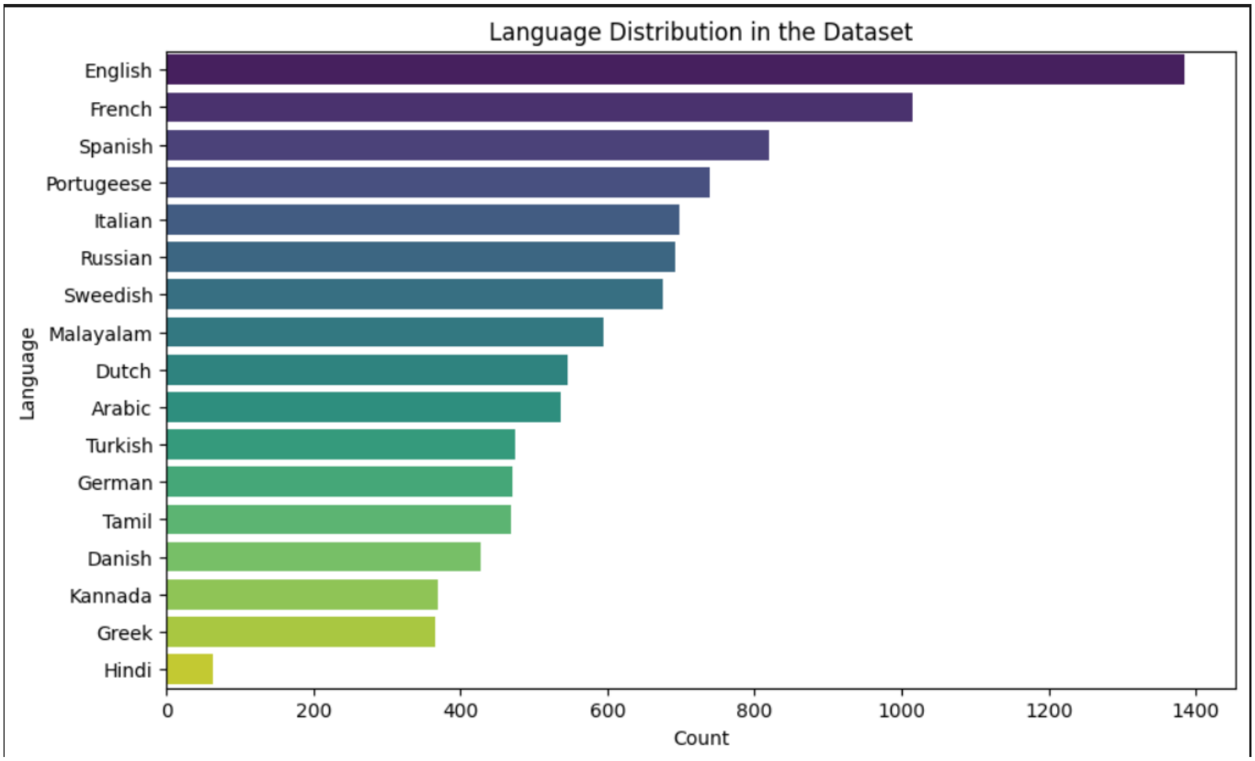
- **Languages Covered**: 17 languages
- **Text Data**: Each sample contains a piece of text in one of the languages.
- **Language Label**: The label indicates the language of the text.

Exploratory Data Analysis

To gain insights into the dataset, the distribution of text samples across the 17 languages was analyzed. Visualizing the language distribution provided a better understanding of the dataset's structure and helped identify any imbalances in the representation of certain languages.

Language Distribution Visualization

The language distribution chart illustrates how frequently each language appears in the dataset, providing a clear picture of the dataset's composition.



5. METHODOLOGY:

Data Preprocessing

The text data underwent several preprocessing steps to prepare it for machine learning models:

1. **Stopword Removal**: Common stopwords (e.g., "the", "and") were removed based on predefined stopword lists for each language.
2. **TF-IDF Vectorization**: The text was transformed into numerical features using **TF-IDF (Term Frequency-Inverse Document Frequency)**, which captures the importance of each word in the text relative to the entire corpus.

Model Selection

Two machine learning models were selected for this task:

- **Naive Bayes**: A simple probabilistic model that is commonly used for text classification tasks due to its efficiency and performance.
- **SVM (Support Vector Machines)**: A more complex model that performs well on high-dimensional data like text, making it a strong candidate for this classification task.

Training and Testing

The dataset was split into training and test sets. Both the Naive Bayes and SVM models were trained on the training data and then evaluated on the test data to assess their performance.

Evaluation Metrics

To evaluate the models, several key metrics were used:

- **Accuracy**: Measures the percentage of correct predictions made by the model.
- **Confusion Matrix**: A detailed view of the model's predictions, showing where it correctly or incorrectly classified languages.
- **Classification Report**: Provides metrics such as precision, recall, and F1-score for each language, offering a more granular evaluation of the model's performance.

6. RESULTS AND ANALYSIS:

Model Performance

After training and testing both models, their performance was evaluated. The **SVM** model provided good accuracy and was efficient in terms of computation, while the **Naive Bayes** model performed slightly better in terms of precision and recall for specific languages.

SVM Results

- **Accuracy**: SVM achieved a competitive accuracy on the test set, showing that it effectively learns language patterns from the data.

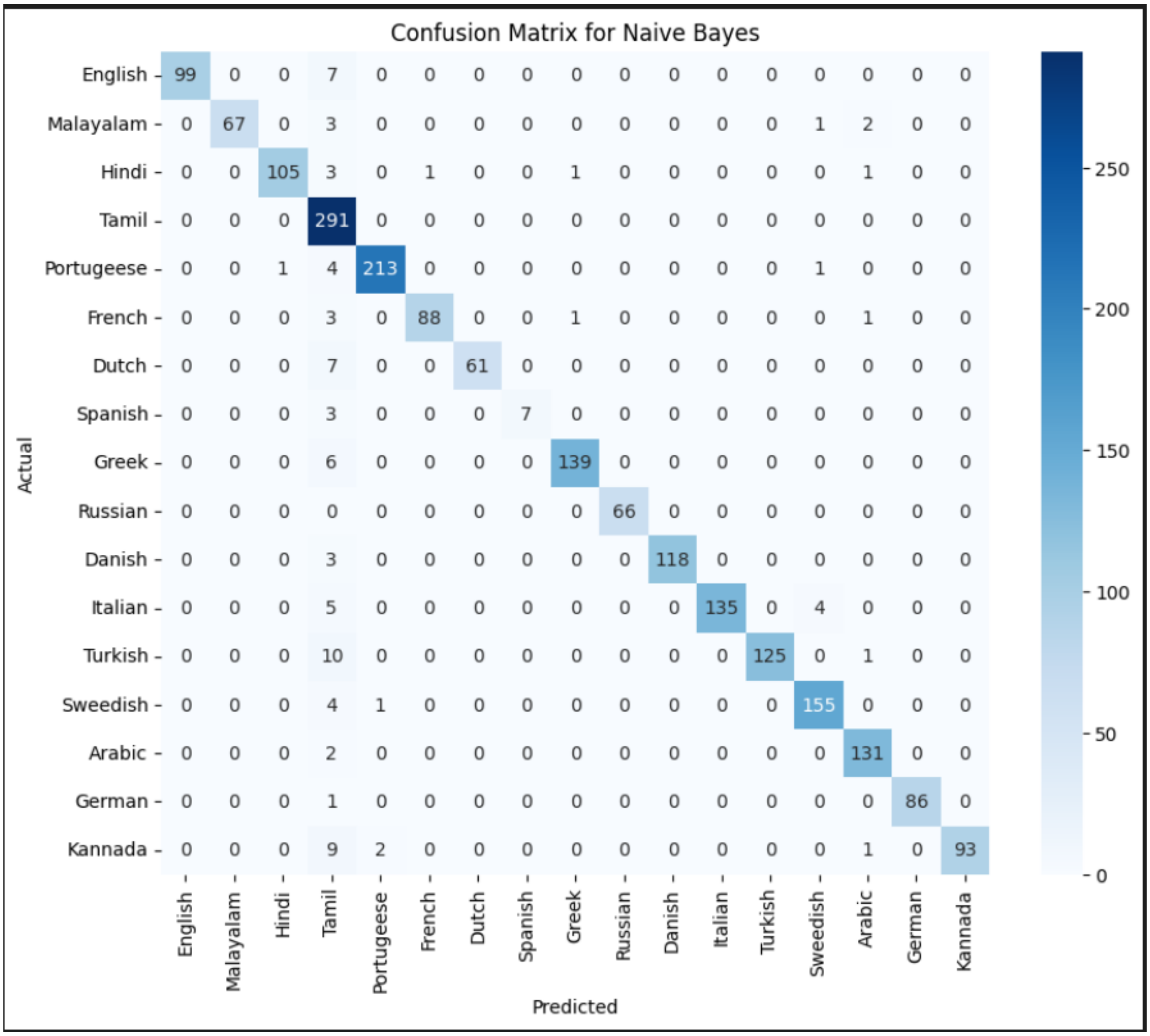
Naïve Bayes Results

- **Accuracy**: Naïve Bayes provided slightly higher accuracy compared to SVM, particularly excelling in the classification of more complex language data.

Confusion Matrix

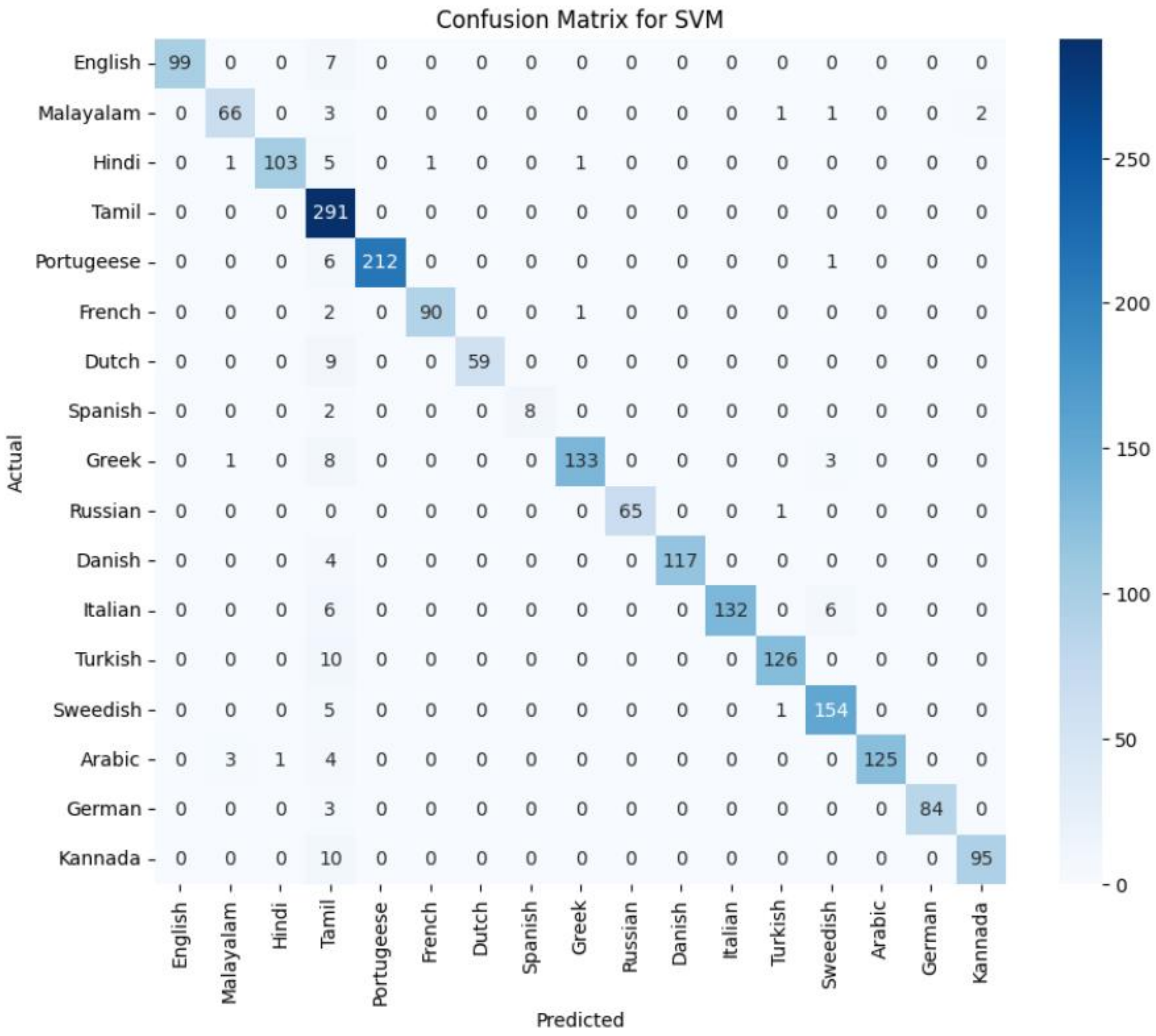
The confusion matrix for both models highlighted areas where they performed well and where they struggled. The matrix provides insights into specific languages that were frequently misclassified and those that were predicted accurately.

Naive Bayes Confusion Matrix



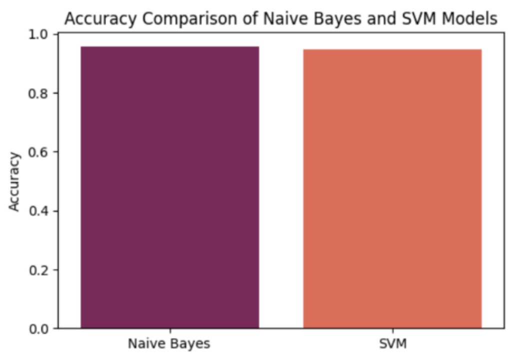
	English	Malayalam	Hindi	Tamil	Portugeese	French	Dutch	Spanish	Greek	Russian	Danish	Italian	Turkish	Sweedish	Arabic	German	Kannada
English	99	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0
Malayalam	0	67	0	3	0	0	0	0	0	0	0	0	0	1	2	0	0
Hindi	0	0	105	3	0	1	0	0	1	0	0	0	0	0	1	0	0
Tamil	0	0	0	291	0	0	0	0	0	0	0	0	0	0	0	0	0
Portugeese	0	0	1	4	213	0	0	0	0	0	0	0	0	1	0	0	0
French	0	0	0	3	0	88	0	0	1	0	0	0	0	0	1	0	0
Dutch	0	0	0	7	0	0	61	0	0	0	0	0	0	0	0	0	0
Spanish	0	0	0	3	0	0	0	7	0	0	0	0	0	0	0	0	0
Greek	0	0	0	6	0	0	0	0	139	0	0	0	0	0	0	0	0
Russian	0	0	0	0	0	0	0	0	0	66	0	0	0	0	0	0	0
Danish	0	0	0	3	0	0	0	0	0	0	118	0	0	0	0	0	0
Italian	0	0	0	5	0	0	0	0	0	0	0	135	0	4	0	0	0
Turkish	0	0	0	10	0	0	0	0	0	0	0	0	125	0	1	0	0
Sweedish	0	0	0	4	1	0	0	0	0	0	0	0	0	155	0	0	0
Arabic	0	0	0	2	0	0	0	0	0	0	0	0	0	0	131	0	0
German	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	86	0
Kannada	0	0	0	9	2	0	0	0	0	0	0	0	0	0	1	0	93

SVM Confusion Matrix

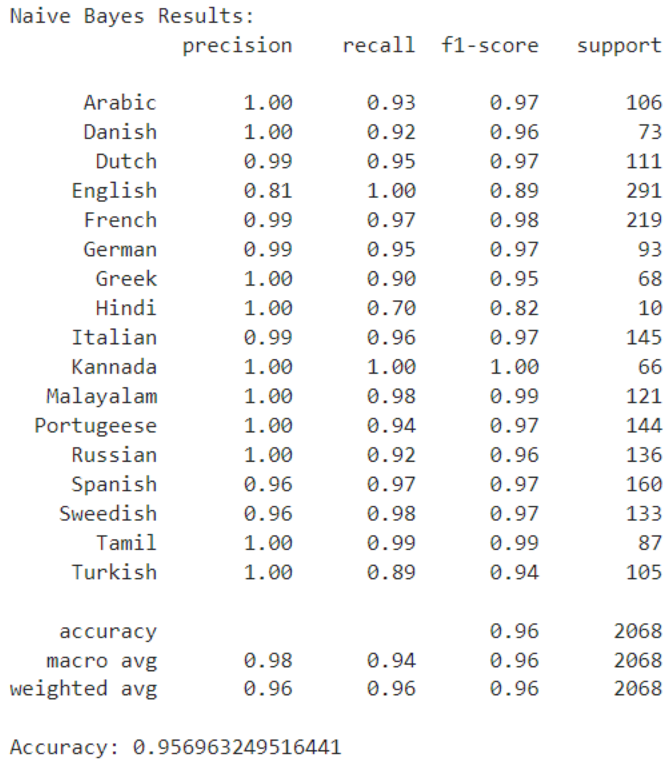


Accuracy Comparison

A comparison between the accuracy scores of the two models revealed that **Naive Bayes** slightly outperformed **SVM,** particularly in handling more challenging language samples.

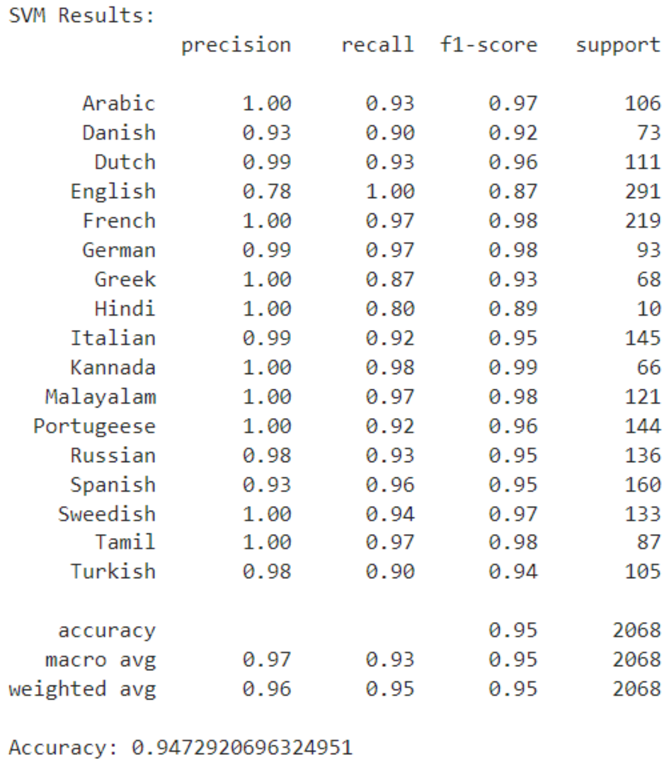


Accuracy of Naive Bayes: 95.6%



```
Naive Bayes Results:
              precision    recall  f1-score   support

      Arabic       1.00      0.93      0.97       106
      Danish       1.00      0.92      0.96        73
       Dutch       0.99      0.95      0.97       111
     English       0.81      1.00      0.89       291
      French       0.99      0.97      0.98       219
      German       0.99      0.95      0.97        93
       Greek       1.00      0.90      0.95        68
       Hindi       1.00      0.70      0.82        10
     Italian       0.99      0.96      0.97       145
     Kannada       1.00      1.00      1.00        66
   Malayalam       1.00      0.98      0.99       121
  Portugeese       1.00      0.94      0.97       144
     Russian       1.00      0.92      0.96       136
     Spanish       0.96      0.97      0.97       160
    Sweedish       0.96      0.98      0.97       133
       Tamil       1.00      0.99      0.99        87
     Turkish       1.00      0.89      0.94       105

    accuracy                           0.96      2068
   macro avg       0.98      0.94      0.96      2068
weighted avg       0.96      0.96      0.96      2068

Accuracy: 0.956963249516441
```

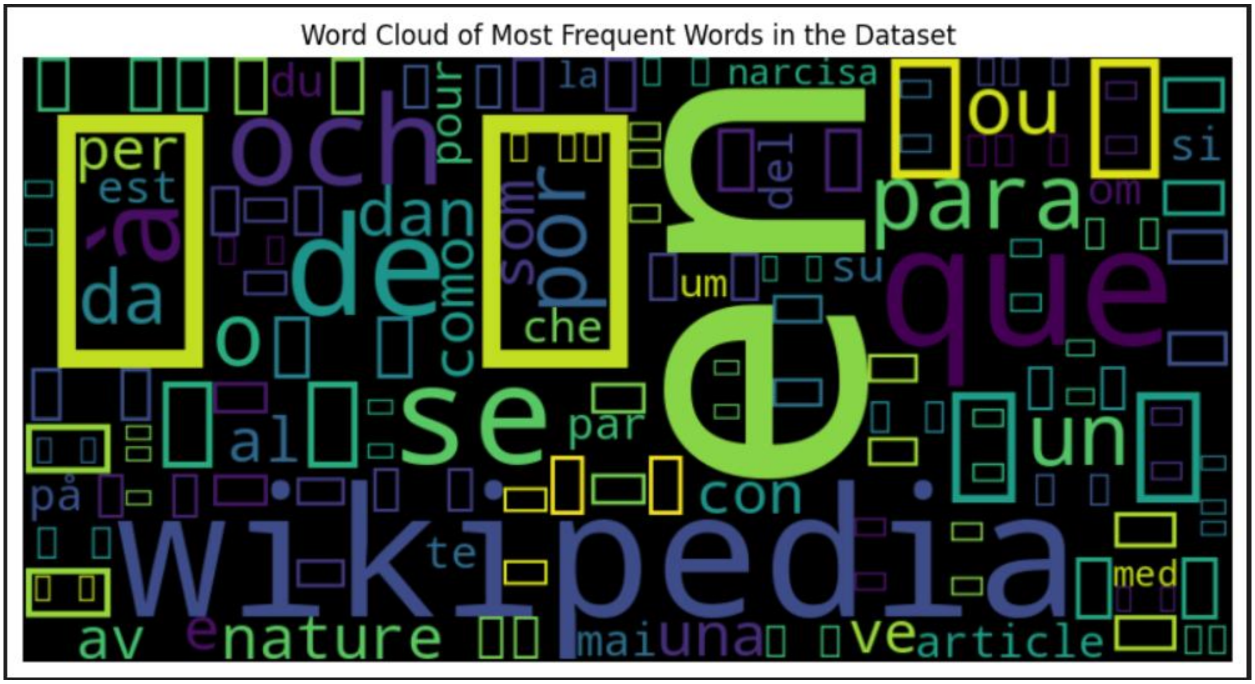
Accuracy of SVM Model: 94.7%



```
SVM Results:
              precision    recall  f1-score   support

      Arabic       1.00      0.93      0.97       106
      Danish       0.93      0.90      0.92        73
       Dutch       0.99      0.93      0.96       111
     English       0.78      1.00      0.87       291
      French       1.00      0.97      0.98       219
      German       0.99      0.97      0.98        93
       Greek       1.00      0.87      0.93        68
       Hindi       1.00      0.80      0.89        10
     Italian       0.99      0.92      0.95       145
     Kannada       1.00      0.98      0.99        66
   Malayalam       1.00      0.97      0.98       121
  Portugeese       1.00      0.92      0.96       144
     Russian       0.98      0.93      0.95       136
     Spanish       0.93      0.96      0.95       160
    Sweedish       1.00      0.94      0.97       133
       Tamil       1.00      0.97      0.98        87
     Turkish       0.98      0.90      0.94       105

    accuracy                           0.95      2068
   macro avg       0.97      0.93      0.95      2068
weighted avg       0.96      0.95      0.95      2068

Accuracy: 0.9472920696324951
```

TF-IDF Word Cloud

A word cloud was generated based on the **TF-IDF** scores, highlighting the most important words across all languages in the dataset. This visualization helps to understand which words contribute the most to the models' decision-making.



7. Conclusion

The project successfully demonstrated that both **Naive Bayes** and **SVM** are effective models for language detection tasks. While **SVM** offers fast training and good performance, **Naïve Bayes** provided better overall accuracy, particularly in distinguishing between similar languages.

Key Takeaways

- **Naive Bayes**: More accurate in this case, effectively identifying language patterns across the dataset. It is efficient and performs well, making it suitable for large-scale text data, especially when quick and reliable training is needed.

- **SVM**: While it requires more computational resources, SVM can still be valuable in distinguishing complex language patterns. However, in this case, Naive Bayes provided higher accuracy, likely due to its simpler probabilistic approach working better for this particular dataset.

Future Work

Further improvements to the system could include:

- **Hyperparameter Tuning**: Optimizing model parameters to further enhance performance.
- **Data Augmentation**: Expanding the dataset to include more diverse language samples, improving the model's robustness.
- **Integration**: Incorporating this language detection system into real-world applications such as automatic translation pipelines.

The language detection system can now serve as a foundational model for tasks requiring accurate identification of languages in text, enabling improvements in various NLP-related applications.