# Language Identification System

## Overview:

This project focuses on building a language identification system. It can accurately detect the language of a given text input. By using machine learning techniques such as Multinomial Naive Bayes, the system has high accuracy and it can be used in areas like translation tools, chatbots, and search engines.

## Features:

\* Supports multiple languages, including English, French, Hindi, Spanish, German, and more.

\* High accuracy (94.7%)

\* Handles short text inputs and mixed-language scenarios.

\* Easy to integrate into various applications

## Prerequisites:

To run the project, ensure you have the following:

\* Python 3.7 or higher

\* Required libraries:

o pandas

o numpy

o scikit-learn

o matplotlib

o seaborn

You can install these dependencies using the command:

*pip install pandas numpy scikit-learn matplotlib seaborn*

## Dataset:

The dataset used for this project contains approximately 10,000 samples of text in 17 different languages. Each sample includes the text and its corresponding language label.

## Dataset Link:

You can download the dataset from Kaggle: Language Detection Dataset

## Steps to Download the Dataset

1. Visit the dataset link above.

2. Download the dataset as a .csv file.

3. Save the dataset in the project directory under a folder named data.

4. Rename the file to language-detection.csv.

## How to Run the System

Follow these steps to run the language identification system:

Step 1: Clone the Repository

Clone this repository to your local machine:

git clone <repository-url>

Step 2: Navigate to the Project Directory

cd language-identification-system

Step 3: Load the Dataset

Ensure the dataset (language-detection.csv) is placed in the data folder.

Step 4: Run the Script

Execute the main script to train the model and test predictions:

python scripts/main.py

Step 5: Predict a Language

You can input a custom text to predict its language. Modify the predict function in the script:

text = "Pickleball"  # Example input

print(predict(text))

Implementation Details

## Data Preprocessing

1. Cleaning

2. Vectorization

3. Label Encoding

4. Splitting

## Machine Learning Model:

We use the Multinomial Naive Bayes (MNB) classifier text classification tasks. The model is trained on vectorized data and evaluated using metrics like accuracy, precision, recall, and F1-score.

## References

\* Dataset: Language Detection Dataset on Kaggle

\* Scikit-learn documentation: https://scikit-learn.org/

\* Python Regular Expressions: https://docs.python.org/3/library/re.html

## Future Improvements

1. Expand Language Support: Add more languages to the dataset.

2. Enhance Preprocessing: Include techniques like stemming, …

3. Experiment with Advanced Models: Try Logistic Regression, SVM, ...

4. Handle Mixed-language Texts: Improve the model’s ability to classify multilingual inputs.