Machine Learning Basics

Instructors: Babak n. Arabi, Mohammadreza A. Dehaqani, Mostafa Tavassolipour

Mostafa Kermani Nia, Faezeh Mozaffari, Mahan Osouli

Fall 2025

Final Project

Introduction

In Phase 1, you gathered a multilingual speech dataset by selecting audio clips from audiobooks. Now, in Phase 2, you will move from collecting data to preparing and exploring it through a structured machine learning process. This phase will teach you important steps in building a speech-based classification system.

You are now going to process the raw audio files to identify language-specific features. There are five main tasks in this phase:

1. Data Cleaning
2. Feature Extraction
3. Classification
4. Clustering
5. Metrics and Evaluation

By completing Phase 2, you'll have experience working with real-world audio data and developing models to classify and group multilingual speech. These skills are crucial for anyone working in speech processing and machine learning.

Project Description

You can access the dataset on eLearn as well as here: Dataset Link

1. Data Cleaning

Now that data collection is complete, the next step is to perform initial preprocessing to prepare the audio data for the next stages of your project. Since your dataset consists of short (about one-minute) audio clips extracted from audiobooks in Spanish, Korean, Italian, and German, it's important to choose the most suitable preprocessing and augmentation techniques to ensure consistent, clean, and diverse audio input.

2. Feature Extraction

To effectively utilize raw audio data collected in the previous stages, it is necessary to extract meaningful features and transform the audio signals into suitable numerical representations that can be used as inputs to machine learning models.

For feature extraction, students may employ well-known and widely used libraries that provide access to time-domain, frequency-domain, or time–frequency features of audio signals, or propose and implement innovative feature extraction methods inspired by existing techniques. The choice of features should be made carefully, as the extracted features will be used in both classification and clustering tasks. Therefore, the selected features should be informative, discriminative, and appropriate for distinguishing between different spoken languages.

Students are required to justify their choice of features in the project report and explain why these features are suitable for the given task.

3. Classification

In this stage, your goal is to train machine learning models to identify the spoken language of the audio clips using the features extracted in the previous step.

(a) Data Splitting

Divide your dataset into Training and Testing sets (80% train, 20% test). It is crucial to use stratified sampling to ensure that the proportion of each language remains balanced in both sets.

(b) Preprocessing

Apply necessary preprocessing steps such as Normalization to your features. You may also use Dimensionality Reduction techniques like PCA or LDA if your feature vector is too large, to improve training speed and performance.

(c) Model Training

Implement and train at least three different classification algorithms. You are free to choose from standard methods taught in the course, such as:

- K-Nearest Neighbors
- Support Vector Machines
- Multi-Layer Perceptron / Neural Networks
- Decision Trees / Random Forest
- Logistic Regression
- Naïve Bayes

Note: You do not need to use Deep Learning frameworks like CNNs or Transformers unless you specifically want to. Standard ML libraries like Scikit-Learn are sufficient.

4. Clustering

In this section, you will analyze the data in an unsupervised manner to see if the audio samples naturally group together based on their language characteristics.

(a) Algorithms

Apply at least two clustering algorithms (K-Means and Hierarchical Clustering/DBSCAN/GMM).

(b) Optimal Number of Clusters

Use methods like the Elbow Method or Silhouette Score to determine the optimal number of clusters (k). Discuss whether the optimal k matches the actual number of languages in your dataset.

(c) Visualization

Visualize the clustering results using dimensionality reduction techniques (t-SNE or PCA) to plot the data points in 2D space. This will help illustrate how well the languages are separated in the feature space.

5. Metrics and Evaluation

Merely running the code is not enough; you must evaluate how well your models perform.

(a) Classification

- Report Accuracy, Precision, Recall, and F1-Score (Weighted/Macro avg) for each model.
- Plot the Confusion Matrix. This is vital for analyzing which languages are being confused with each other.
- Compare the performance of the three models you trained.

(b) Clustering

- Report the Silhouette Score.
- Analyze the Cluster Purity or simply discuss the composition of each cluster.

Report & Code

Report

In addition to the correct code, a complete report with thorough explanations, data analysis, and interpretation of graphs and charts holds significant importance. The report is the main source for validating the methods you applied and assessing your understanding of the topic. Thus, correct code without an appropriate report is meaningless. Therefore, ensure that all relevant points are included in the report.

The final report should include all parts that are completed after the submission of the initial report. The deadline for submitting both the report and code is end of 25th Bahman.

Key points to address in your report:

1. A brief explanation of the preprocessing methods you used and the reasons for selecting them.
2. A brief explanation of the features you used and the reasons for selecting them.
3. A brief explanation of why you chose the classification and clustering models used in the project.
4. Mention and analyze the evaluation metrics (Accuracy, F1-Score, Confusion Matrix) for each of the classification models. Compare the models and explain why one performed better than the others.
5. For clustering, discuss the optimal number of clusters found. Visualize the clusters using PCA/t-SNE and analyze whether the clusters correspond to the actual languages.

The report should be well-organized, divided according to the questions outlined in the project description, and contain detailed and precise analyses.

Code

The code should be submitted in a Jupyter Notebook (`.ipynb`) format and must be pre-executed. Clearly separate different sections of the code using cells, and ideally, include titles or headers to make the code more understandable.

Follow the structure below for your code submission:

- `Data_Cleaning_and_Feature_Extraction.ipynb`
- `Classification.ipynb`
- `Clustering.ipynb`
- `Evaluation.ipynb`

Each notebook should contain all the analyses performed in the report, ensuring that the code reflects the steps discussed in the written part.

Final Notes

- Any similarity in project work between individuals in different groups will not be accepted. In the event of any detected plagiarism, actions will be taken according to the rules.
- Using references and artificial intelligence is allowed and even recommended. However, if your report contains translated or directly copied text from these sources, or if you have used reports from other individuals, your work will be considered plagiarism.
- After carefully reviewing these instructions, if you have any questions about the project, it is better to ask them in the course forum on eLearn so that others can benefit from the answers as well. If not, you can either ask in the Telegram group or email the project designers.

- **Contacting Teaching Assistants Responsible for the Project**

  Mostafa Kermani Nia: Mkermani1383@gmail.com, Kermaninia@ut.ac.ir

  Faezeh Mozaffari: faezehxmozaffari@gmail.com, faezehmozaffari@ut.ac.ir

  Mahan Osouli: mahan.osouli@gmail.com, mahan.osouli@ut.ac.ir

Considering the current circumstances, our four-person project team fully understands the situation and will make every effort to ensure that students are able to submit their projects most fairly and equitably.

We are hopeful for brighter days ahead.