BDM 3035 - Big Data Capstone Project

MILESTONE REPORT 03 FOR
SPEECH EMOTION RECOGNITION PROJECT

SUBMITTED TO

Instructor Meysam Effati

SUBMITTED BY

ADRIANA M. PENARANDA BARON (C0898944)

ARUNA GURUNG (C0896129)

CARLOS A. REY PINTO (C0868575)

HALDO J. SOMOZA SOLIS (C0904838)

PUJAN SHRESTHA (C0901167)

July 10, 2024

Table of Contents

I. INTRODUCTION

The Speech Emotion Recognition (SER) project focuses on creating a Python-based model that can detect human emotions from speech using the Librosa library and machine learning techniques. This application is particularly useful for call centers, where recognizing customer emotions can improve service quality and boost conversion rates. By leveraging the dataset, which includes emotionally labeled speech and song files, we will extract crucial audio features using Librosa, train a machine learning model, and assess its accuracy.

This project provides practical experience in audio processing and machine learning, showcasing the effective integration of these technologies in understanding human emotions. By carrying out this project, we will gain hands-on experience in audio processing, feature extraction, and machine learning model training, offering a thorough understanding of SER systems.

The GitHub link were is uploaded the milestones notebooks is:

https://github.com/adripenaranda/Speech_Emotion_Recognition_Project

II. PROGRESS REPORT

SUMMARY OF TASKS COMPLETED

In this step on the first place we checked if the dataset was balance to try to improve our model performance, we realized the data was very lightly unbalance but still we decided to applied SMOTE to balance class distributions and used GridSearchCV for hyperparameter tuning of an MLP classifier. However, the model performed better without SMOTE, so we finally decided to leave the same model we had before, just improving the hyperparameters alpha to 0,001 and hidden layers to 500 we could improve our accuracy to 79,96%.

Also We developed a function to generate and visualize spectrograms for different emotions and we analyzed them to gain insights.

KEY ACHIEVEMENTS AND MILESTONES REACHED

These are the achievements and milestones reached

Milestone 1:

- Gather the dataset.
- Explore and understand the dataset structure.
- Extract audio features using Librosa.
- Split the data into training, validation, and test sets.
- Select appropriate machine learning model for emotion recognition.
- Implement initial model using data.
- Train model on the training data
- Test model on the validation set.
- Evaluate initial model performance and identify areas for improvement.

Milestone 2:

- Fine-tune model hyperparameters.
- Implement additional feature extraction techniques if needed.
- Retrain and test models with refined parameters.
- Perform cross-validation to ensure model robustness.
- Compare performance metrics and select the best model.

Milestone 3:

- Check data balance
- Finish tuning model
- Create plots of spectrograms for different emotions.
- Visualize spectrograms to analyze frequency patterns related to different emotions.
- Document findings from spectrogram analysis.
- Integrate spectrogram visualization into model evaluation if it is needed.

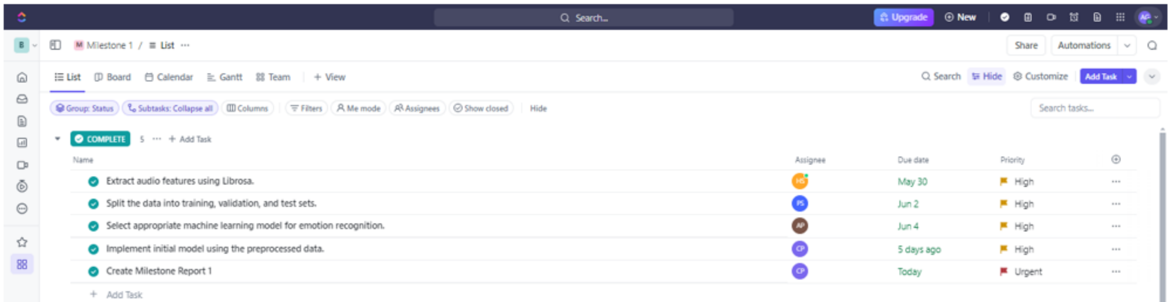
DEVIATIONS FROM THE ORIGINAL PLAN AND REASONS FOR THESE CHANGES.

We didn't face any issue about the development of this milestone, just was surprised that balancing the dataset we didn't get a better performance for the model. The plan is the same for the rest of the project.

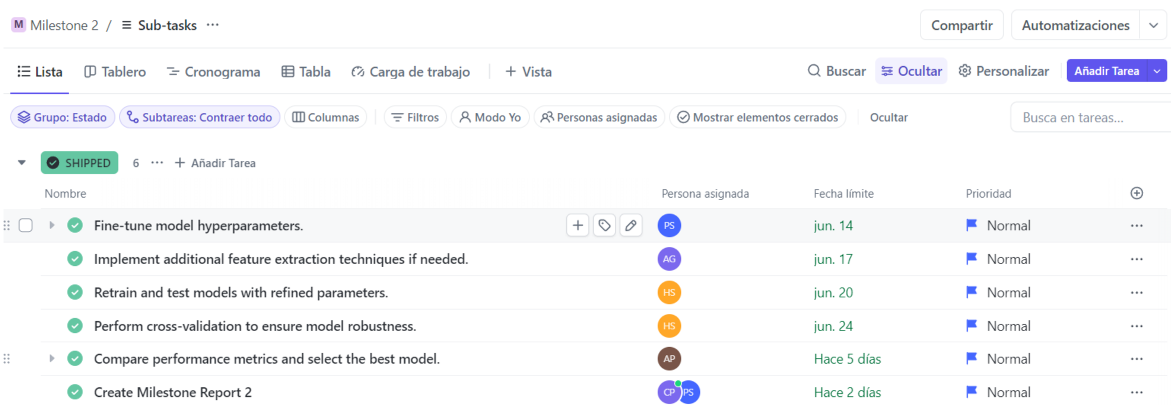
III. MODIFIED TIMELINE TABLE

After the development of this step we don't have to modify the timeline table for the up coming steps.

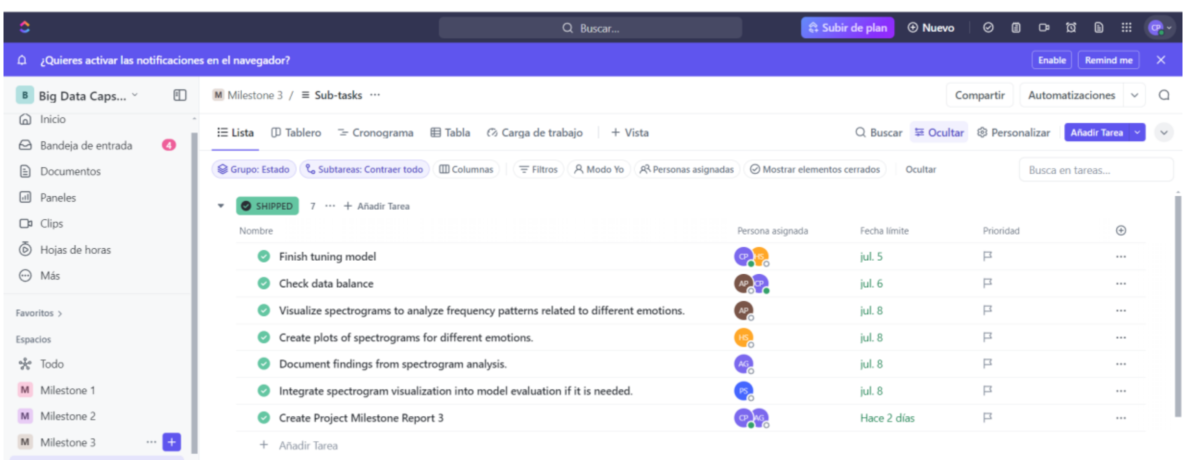
MILESTONE 1



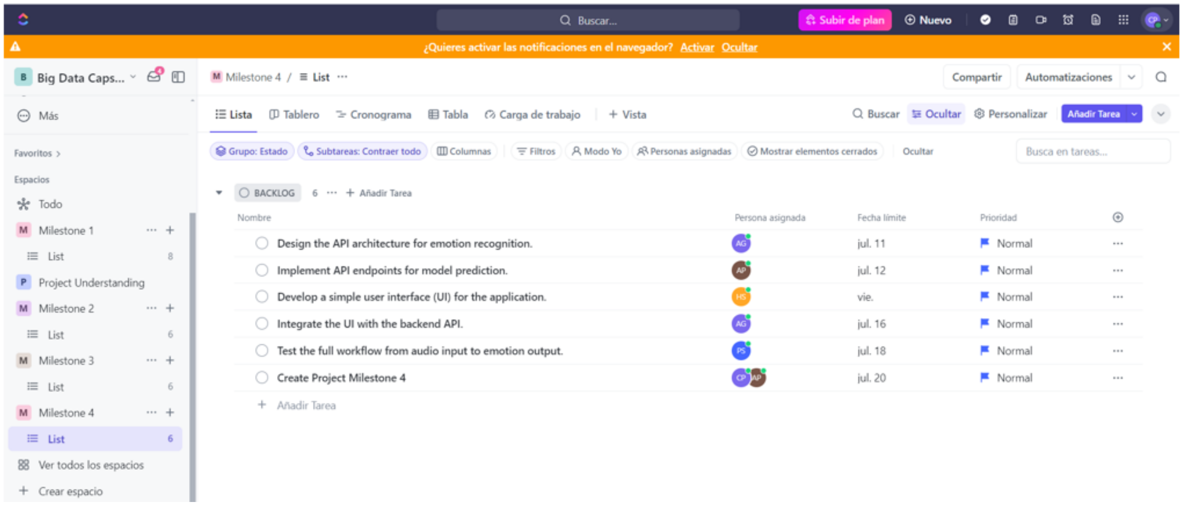
MILESTONE 2



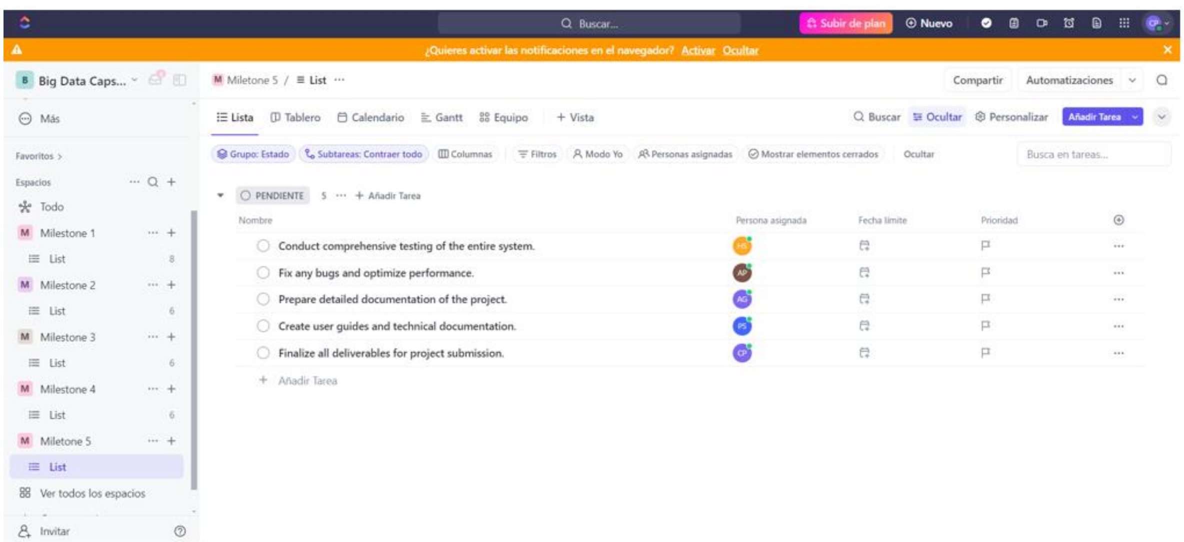
MILESTONE 3



MILESTONE 4



MILESTONE 5



IV. NEXT STEPS

UPCOMING STEPS AND ACTIVITIES PLANNED FOR THE PROJECT.

Brief Description of Tasks to be Undertaken

Milestone 4: July 20, 2024 - API development and UI integration.

1. **Design the API architecture for emotion recognition**

    - **Explanation**: Create a structured blueprint for the API that will handle requests and responses for emotion recognition.
    - **Purpose**: Ensure a scalable and efficient system to process audio inputs and return emotion predictions.

2. **Implement API endpoints for model prediction**

    - **Explanation**: Develop the actual API endpoints that will receive audio files, process them, and return emotion predictions using the trained model.
    - **Purpose**: Enable external applications to interact with the emotion recognition model through a standardized interface.

3. **Develop a simple user interface (UI) for the application**

- **Explanation**: Create a basic UI that allows users to upload audio files and view the predicted emotions.
- **Purpose**: Provide an accessible and user-friendly way for users to interact with the emotion recognition system.

4. **Integrate the UI with the backend API**

- **Explanation**: Connect the front-end UI with the backend API so that user inputs are processed and results are displayed seamlessly.
- **Purpose**: Ensure a smooth and functional user experience from audio input to emotion prediction output.

5. **Test the full workflow from audio input to emotion output**

- **Explanation**: Conduct thorough testing of the entire system, from audio file submission through the UI to receiving emotion predictions from the API.
- **Purpose**: Validate the reliability and accuracy of the integrated system to ensure it meets user expectations and performs as intended.

V. CHALLENGES FACED

SMOTE Implementation: After applying SMOTE to balance the class distributions, we observed SMOTE was didn’t reach the expected improvement for the model. Giving a performance very similar that we already had. This suggested that the synthetic samples might have introduced noise.

Model Complexity: Finding the optimal hyperparameters for the MLP classifier was computationally intensive and required careful tuning to avoid overfitting.

Spectrogram Analysis: Interpreting the spectrograms for different emotions was challenging due to the variability in the audio data and the subtle differences between certain emotions.

VI. LESSONS LEARNED

1. **Balancing Techniques**: While SMOTE is a powerful tool for balancing datasets, it is not always the best choice. In our case, it negatively impacted model performance, highlighting the importance of evaluating different balancing techniques and their effects on the model.
2. **Model Evaluation**: Cross-validation is crucial for robust model evaluation. It helped us understand the variability in model performance and ensured that our findings were consistent across different data splits.
3. **Visualization**: Visualizing spectrograms provided valuable insights into the frequency patterns associated with different emotions.

IV. CONCLUSSION

1. **Optimal Model Performance**: After extensive tuning and evaluation, we determined that applying SMOTE was not beneficial for our specific dataset. The best model performance was achieved without applying SMOTE, indicating that the original class distributions were sufficient for training an effective model just improving alpha and hidden layers.
2. **Spectrogram Insights**: The spectrograms revealed distinct frequency patterns for each emotion, which aligned with our expectations. These visualizations can aid in further refining the feature extraction process and improving model accuracy.
3. **Future Work**: Future efforts should focus on exploring other balancing techniques, such as undersampling or advanced synthetic data generation methods. Additionally, integrating more robust feature extraction methods and leveraging advanced deep learning architectures may further enhance model performance.