Ironhack

EMOSENSE

Enia Lahcene
Benjamín
Daniela García

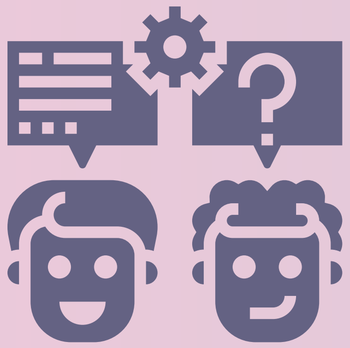
March 11, 2025

PRESENTATION STRUCTURE

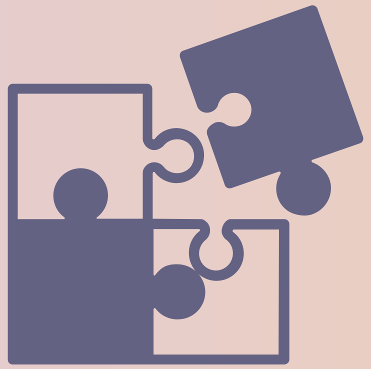
Our agenda today

- Main goals
- Datasets
- Train and test
- Streamlit
- Conclusions
- Future improvements

MAIN GOALS



Enhancing automated models to boost customer satisfaction with service experience



Embracing the challenge: tackling audio dataset complexity

DATASETS

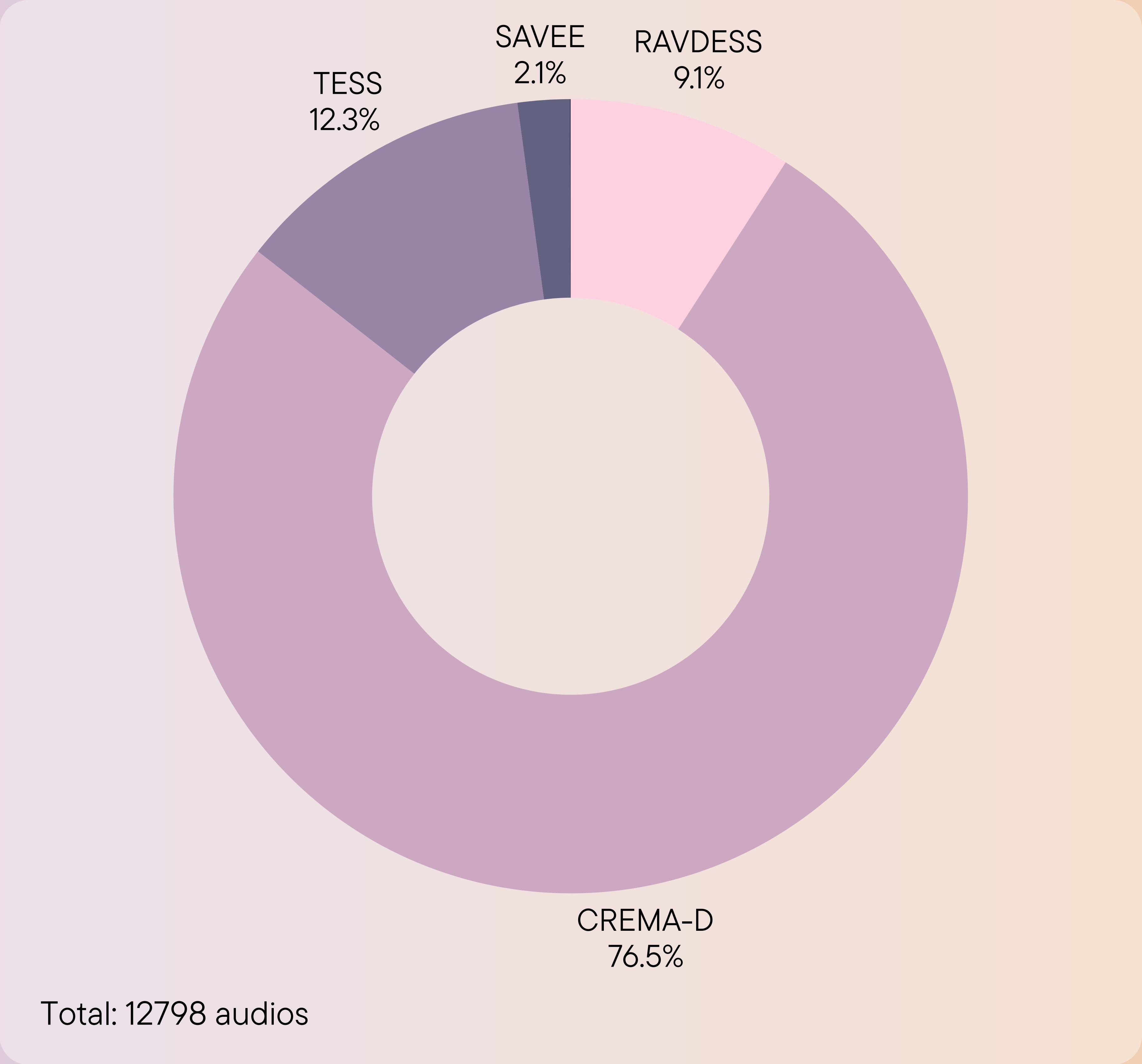
AUDIO DATASETS

TEXT DATASETS

DATASETS

Audio datasets

- Crowd-sourced Emotional Mutimodal Actors Dataset (Crema-D)
- Ryerson Audio-Visual Database of Emotional Speech and Song (Ravdess)
- Surrey Audio-Visual Expressed Emotion (Savee)
- Toronto emotional speech set (Tess)

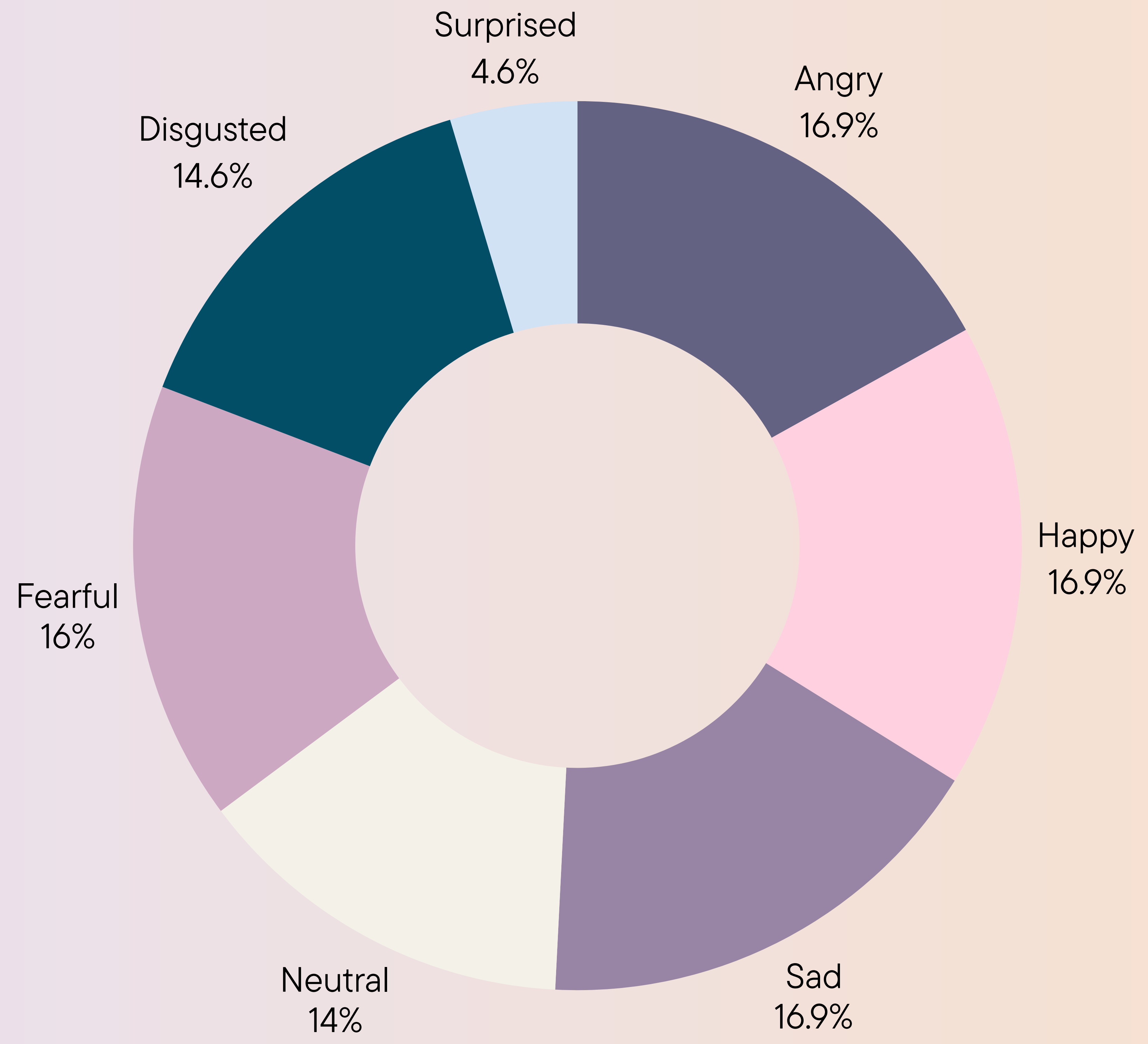


DATASETS

Audio datasets

The chosen emotions are:

- Happy
- Angry
- Neutral



Total: 12798 audios

DATASETS

Text dataset

Emovent contains 8,409 annotate tweets written in Spanish. It is based on events that took place in April 2019 related to different domains: entertainment, catastrophe, political, global commemoration, and global strike.

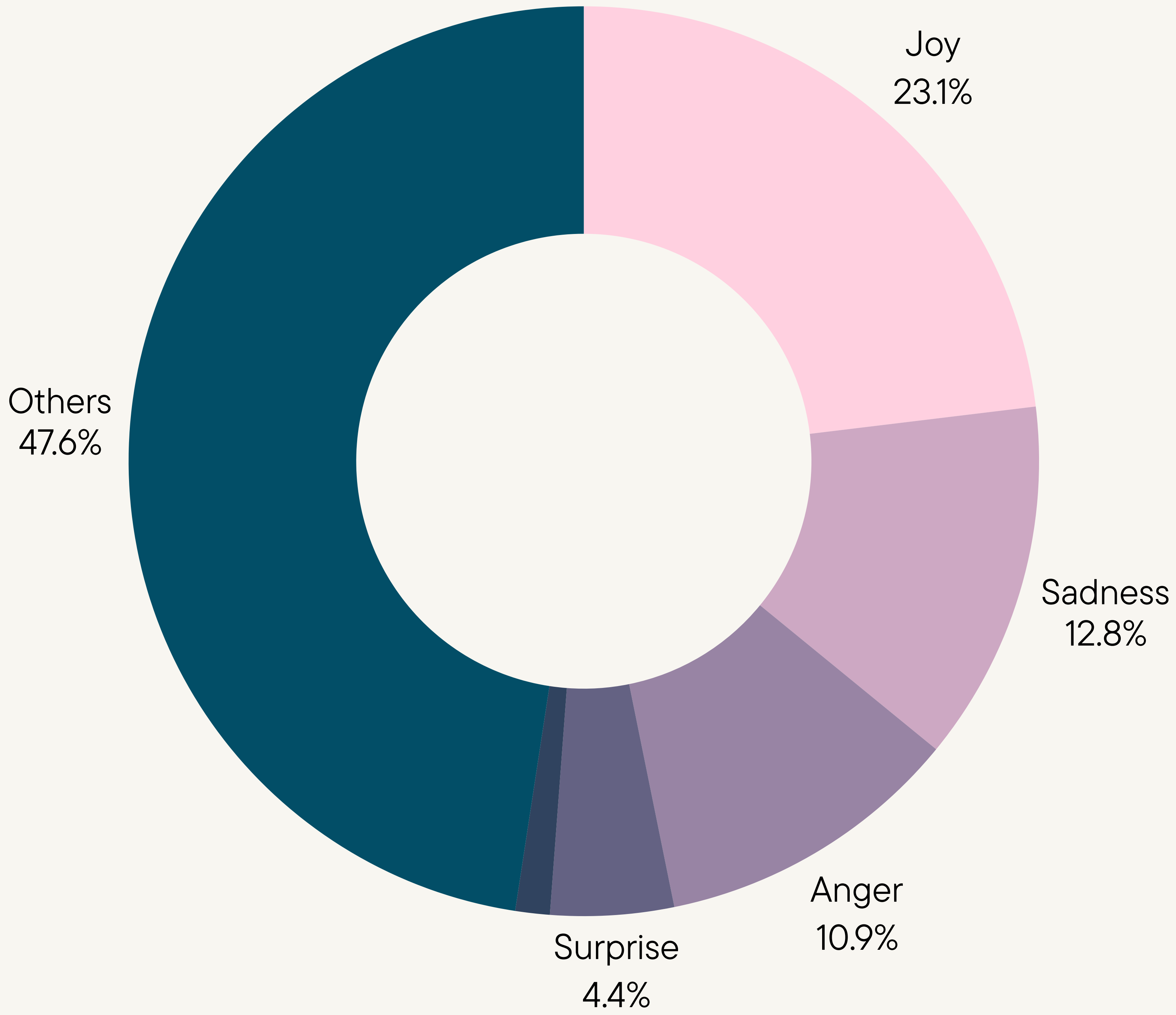
- Anger
- Disgust
- Fear
- Joy
- Sadness
- Surprise

DATASETS

Text dataset

The chosen emotions are:

- Joy
- Anger
- Others



TRAIN AND TEST

Steps we have followed to train and test the models.

TEXT MODEL

STEP 1 Data Processing and Preparation

STEP 2 Model Training

STEP 3 Model Evaluation and Improvement

TEXT DATA PROCESSING AND PREPARATION

Anger → Angry

Joy → Happy

Others → Neutral

- Renamed and standardized labels.
- Filtered only relevant emotions (happy, angry, neutral).
- TF-IDF (Term Frequency-Inverse Document Frequency) to transform text into numerical values.
- SMOTE used to handle class imbalance.

TEXT MODEL TRAINING.

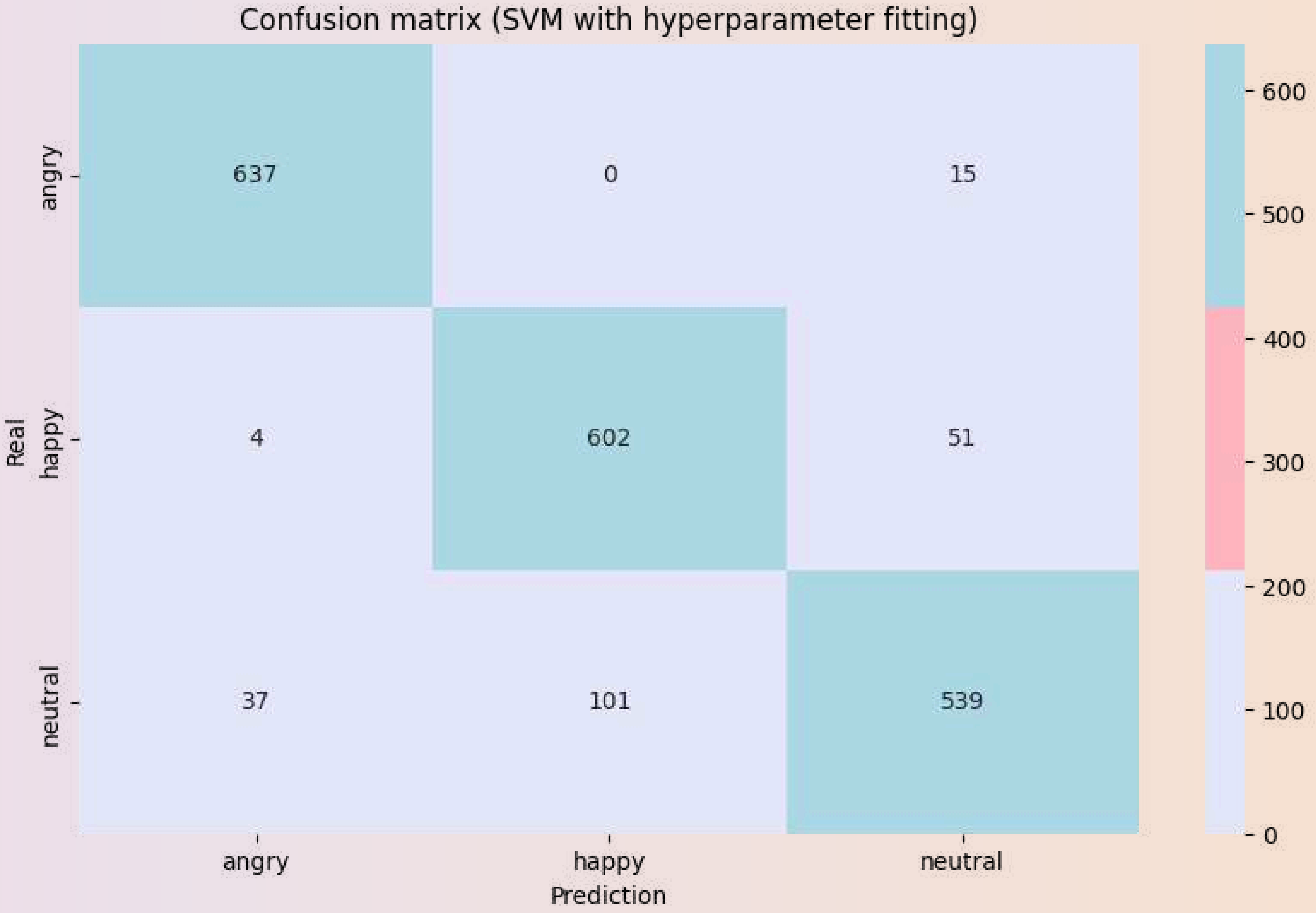
Using SVM Model

Why SVM?	SVM (Support Vector Machine) works well with text classification Finds the best decision boundary
Optimization	Used GridSearchCV to fine-tune hyperparameters Explored different values of C and gamma

TEXT MODEL RESULT

The accuracy obtained for the SVM model with hyperparameter fitting:

93%



AUDIO MODEL

STEP 1 Feature Extraction

STEP 2 Data Processing and Preparation

STEP 3 Model Training

STEP 4 Model Evaluation and Improvement

FEATURE EXTRACTION

Utilized the librosa library to extract key audio features.

MFCC (Mel Frequency Cepstral Coefficients)

Captures the power spectrum of sound and is crucial for distinguishing emotional tonal differences.

Chroma Features

Represents the pitch content and captures harmonics.

Mel Spectrogram

Provides a visual representation of the spectrum of frequencies in the sound.

AUDIO DATA PROCESSING AND PREPARATION

- Extracted features for each audio file.
- Labelled emotions based on predefined categories.
- Encoded labels numerically using LabelEncoder.
- Split dataset using train and tests split to ensure balanced training and testing sets.

MODEL TRAINING

MLP Classifier Setup	Selected Multi-Layer Perceptron (MLP) due to its robust performance with complex data. Configured with hidden layers and adaptive learning rate to accommodate variable data patterns
Training Process	Trained the model on the extracted features to predict emotion labels.

MODEL EVALUATION

Used accuracy score to quantify the model's predictive capabilities.

83%

Achieved robust initial results.

MODEL IMPROVEMENT

SVM Optimization

- Implemented Grid Search (GridSearchCV) to fine-tune Support Vector Machine (SVM) parameters.
- Tested various C, gamma, and kernel parameters for optimal SVM performance.

MODEL IMPROVEMENT

CNN Optimization	Converted audio signals into spectrograms for CNN processing. Designed a CNN architecture with Conv2D, MaxPooling, Dense, and Dropout layers. Trained the model for 20 epochs, optimizing with Adam and sparse categorical cross-entropy.

MODEL COMPARISON

Model	Accuracy
MLP Classifier	83%
SVM	78%
CNN	70%

DEPLOYMENT PREPARATIONS

MODEL SAVING

Utilized joblib for model persistence, saving computational resources by avoiding retraining.

LOADING MECHANISM

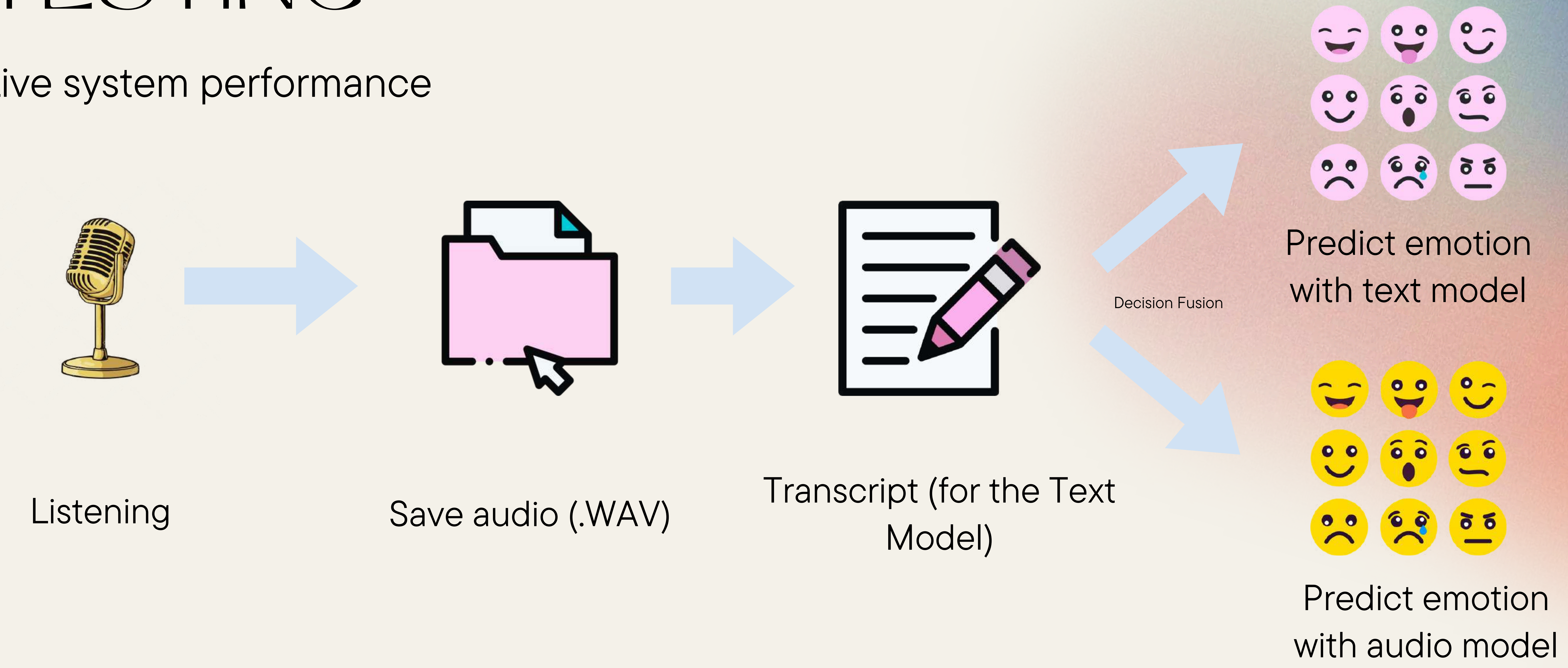
Ensured models could be loaded seamlessly for future predictions and deployment scenarios.

AUDIO CAPTURES AND TEXT CONVERSION

- The microphone is activated with sr.Microphone().
- **adjust_for_ambient_noise** reduces background noise.
- **r.listen(source)** captures the audio.
- recognize_google(audio, ***language***="es-ES") transcribes the audio to text.

MODEL FUSION AND LIVE TESTING

Live system performance



EMOTION FUSION AND FINAL DECISION

Text and audio emotions are combined using rules such as:

- Happy Text 😊 + Angry Audio 😡 → Sarcasm 🙂
- Neutral Text 😐 → Prioritize audio emotion

STREAMLIT

CONCLUSIONS

Text-based emotion recognition has proven to be highly effective.

Audio-based emotion recognition presents unique challenges:

- Finding suitable datasets in Spanish
- Long training times due to the large size of the audio files
- Carefully select the model architecture
- Data quality, diversity and high complexity play a crucial role

In the weighting of the emotion decision, the prediction with the text has more weight.

FUTURE IMPROVEMENTS

- Expand training data to Spanish audio.
- Integrate English text-based emotion recognition
- Real-time feedback loop for model improvement
- Data augmentation to enhance the model
- More emotions
- The emotion decision will be weighted by the model's accuracy.

THANK YOU FOR LISTENING!

Feel free to ask any questions.