

---

# Sprint 3

## Multilingual Subtitle System

**By:** Nimrod Eshel

Welcome to my presentation, let's dive deep into it!



# Table of contents

**01**

## **Introduction**

Why I chose this project

**02**

## **Techniques**

Info extraction, EDA,  
Preprocessing

**03**

## **Basic Models**

Logical Reg, Random  
Forest, XGBoost

**04**

## **Adv Models**

Generic CNN, VGG

**05**

## **Next Steps**

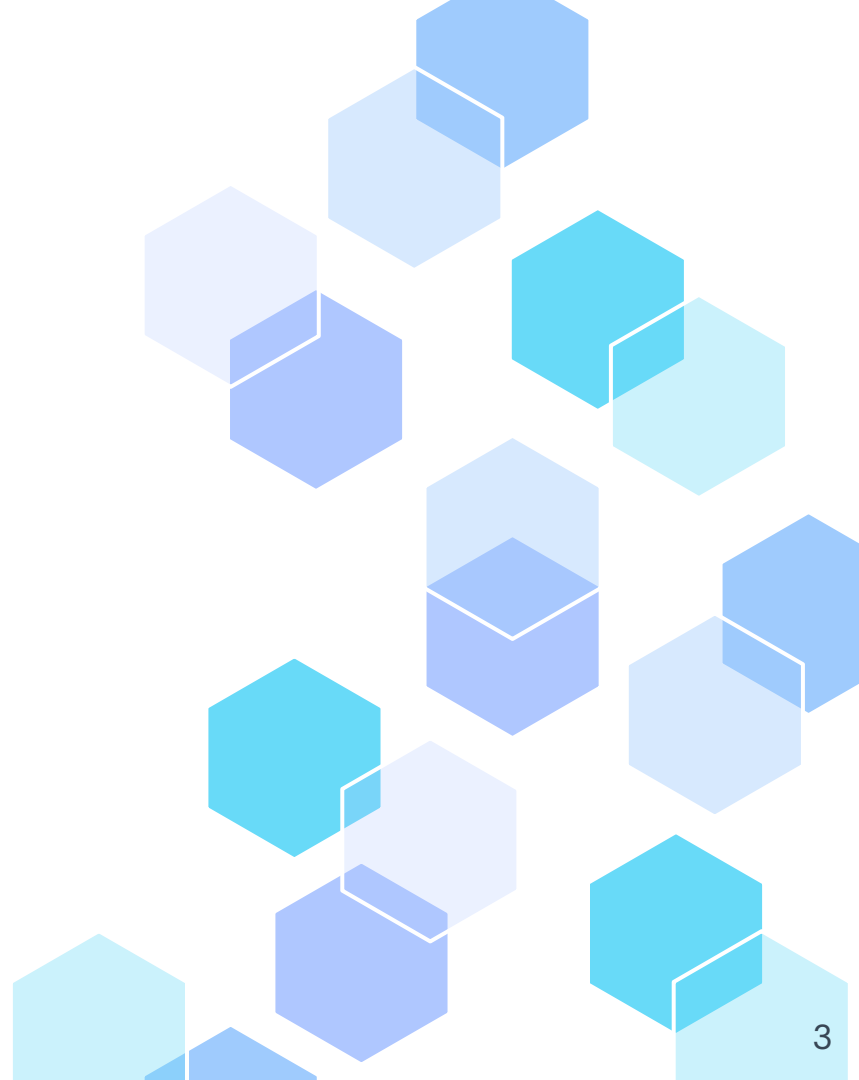
Other CNN Models, UI,  
DevOPs

---

# 01

# Introduction

Why I chose this project?



# Introduction

## What

- I love movies, tv shows, youtube, etc!
- Sometimes movies lack subtitles

## Why

- Absence of subtitles limits media access for non-native speakers
- Ensure everyone enjoys media regardless of the language spoken.

## How

- Detects the audio language, transcribes it, and translates it into subtitles.
- Use Machine learning models to detect language spoken

## Potential Impact:

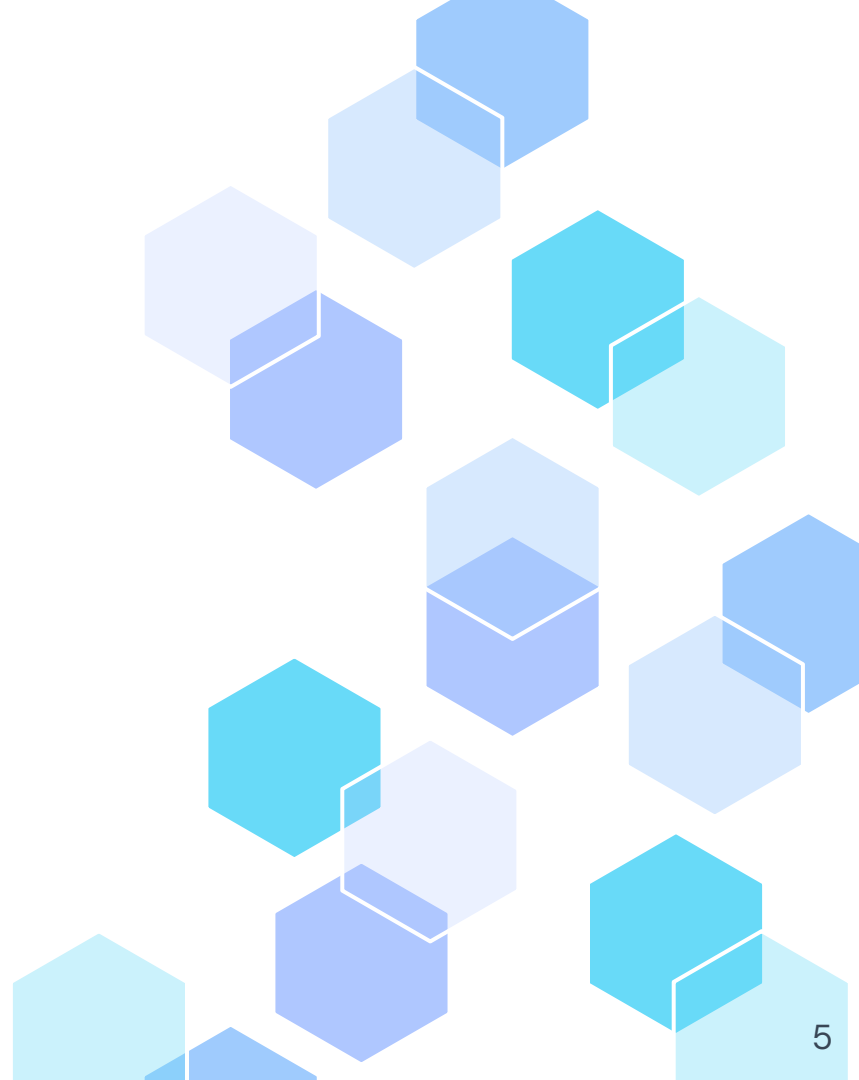
- Enhances accessibility for non-native speakers and the hearing impaired.
- Facilitates language learning and understanding.

---

# 02

# Techniques

Extraction, EDA, Preprocessing



# Extraction

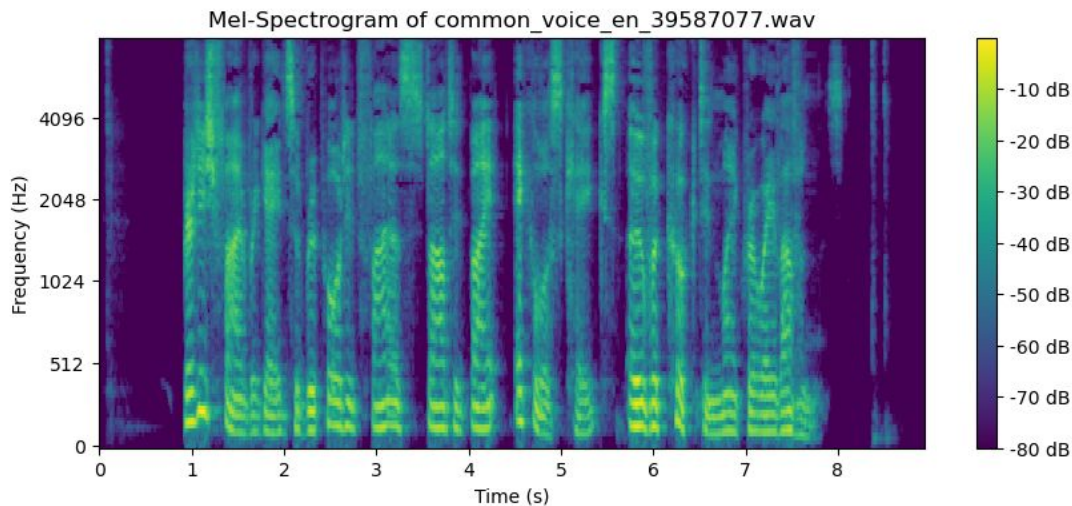
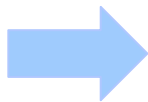
- Extracted audio files for Chinese, English, Spanish, Arabic
- Converted to .wav for better audio quality
- Converted .wav to spectrograms for models
- Resized spectrograms to 128x128
- Flattened pixels from spectrogram to feed models



# Data collection techniques

Common Voice

[moz://a](https://commonvoice.mozilla.org/en/a)



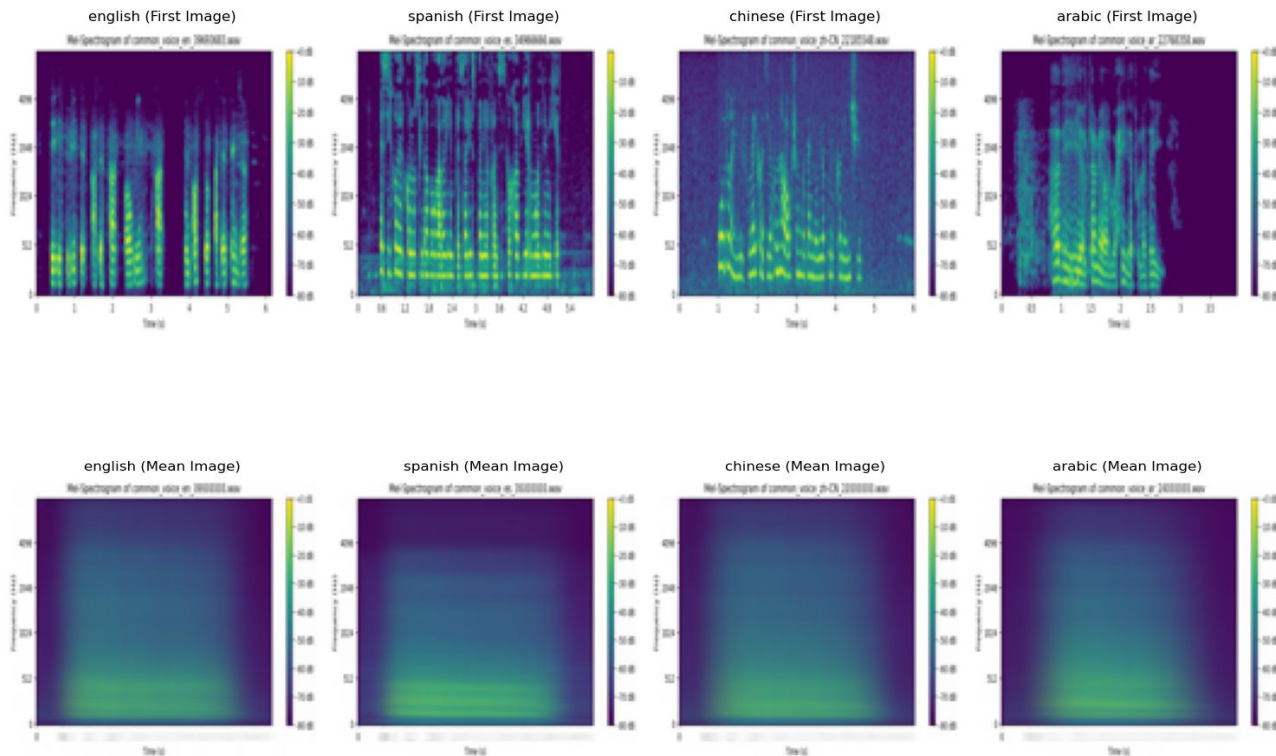
# EDA (Exploratory Data Analysis)

- Reshaped images
- Made figures to showcase uniqueness of each language
- Made separate figures for each language on their average pixel values

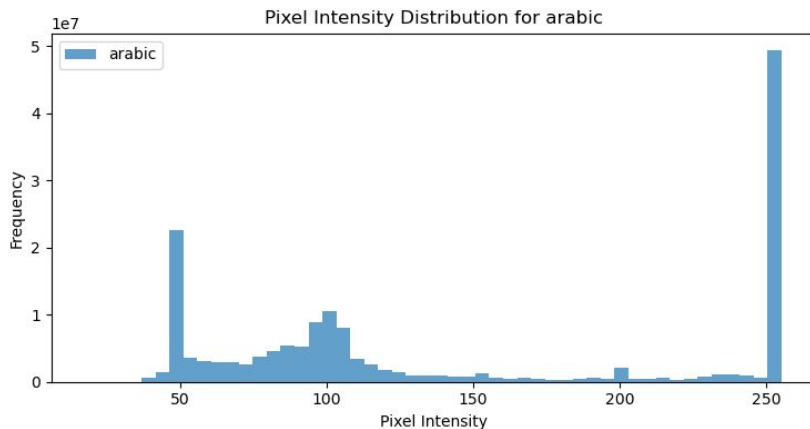
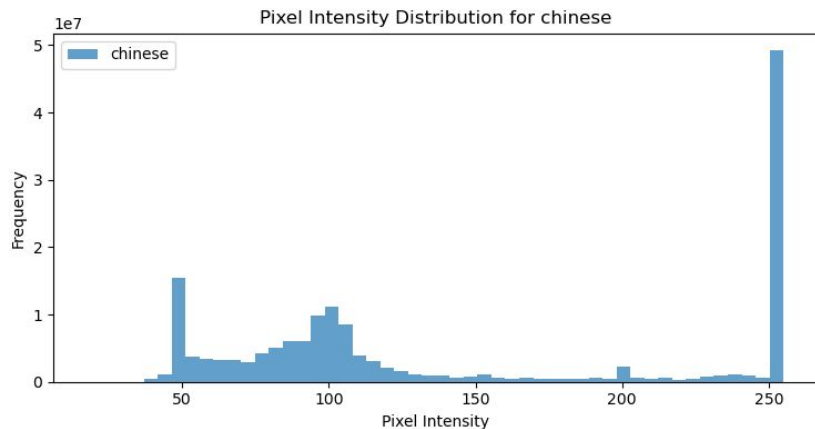
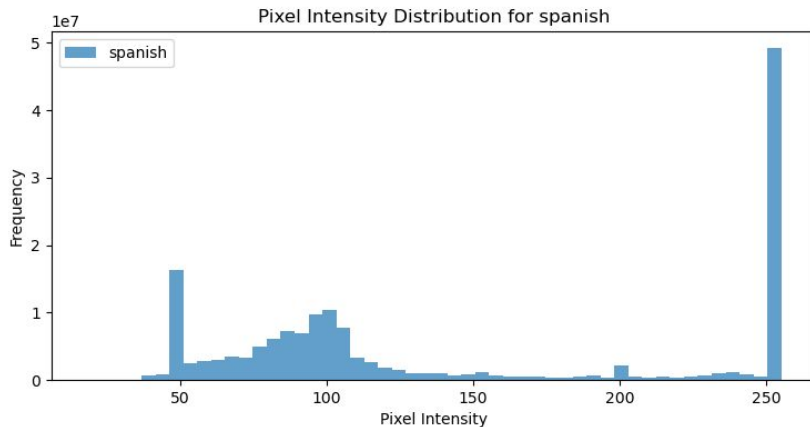
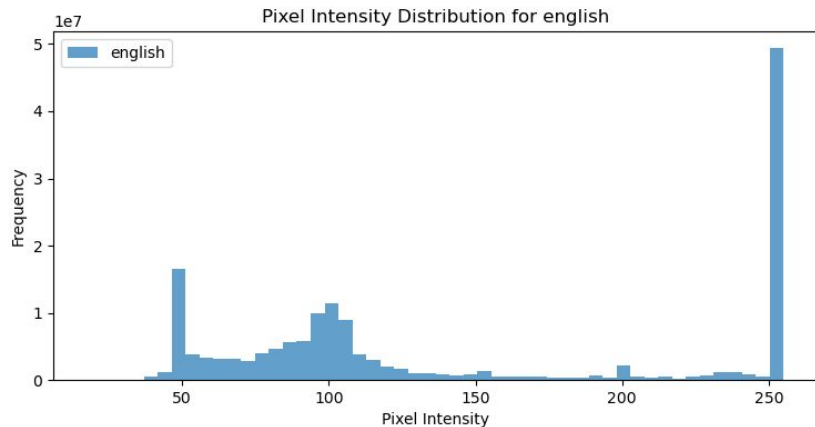




# EDA (Exploratory Data Analysis)



# EDA (Exploratory Data Analysis)



# Preprocessing

- Normalized the flattened images with standard scaling
- Reduced dimensionality with PCA
- Label encoded my languages

But this is all mambo jumbo... all you need to this that this is done before adding my data to my models to make it more accurate.

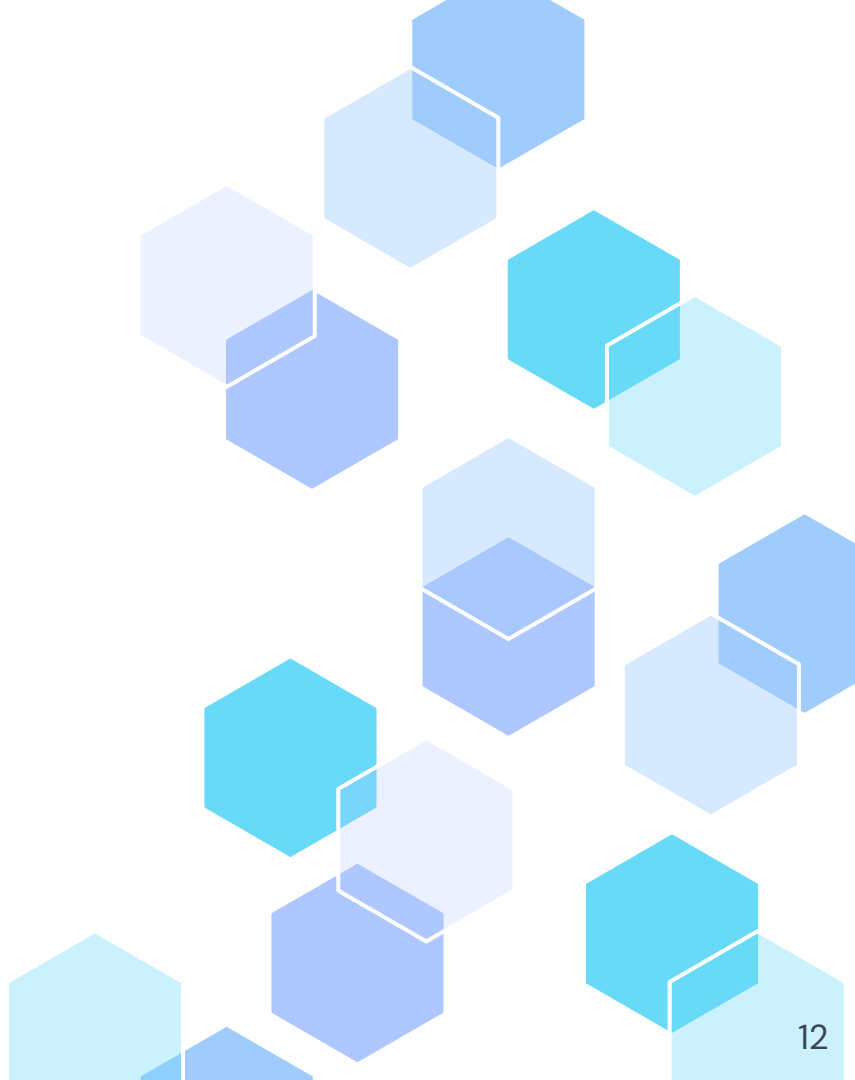


---

# 03

# Models

Cool machine learning models!



# Model Info

I used 3 models (Logistic reg, Random Forest, XGBoost)

## Logistic Regression

- Train Accuracy: 98.4%, Test Accuracy: 97.9%
- 98% Precision & 98% Recall

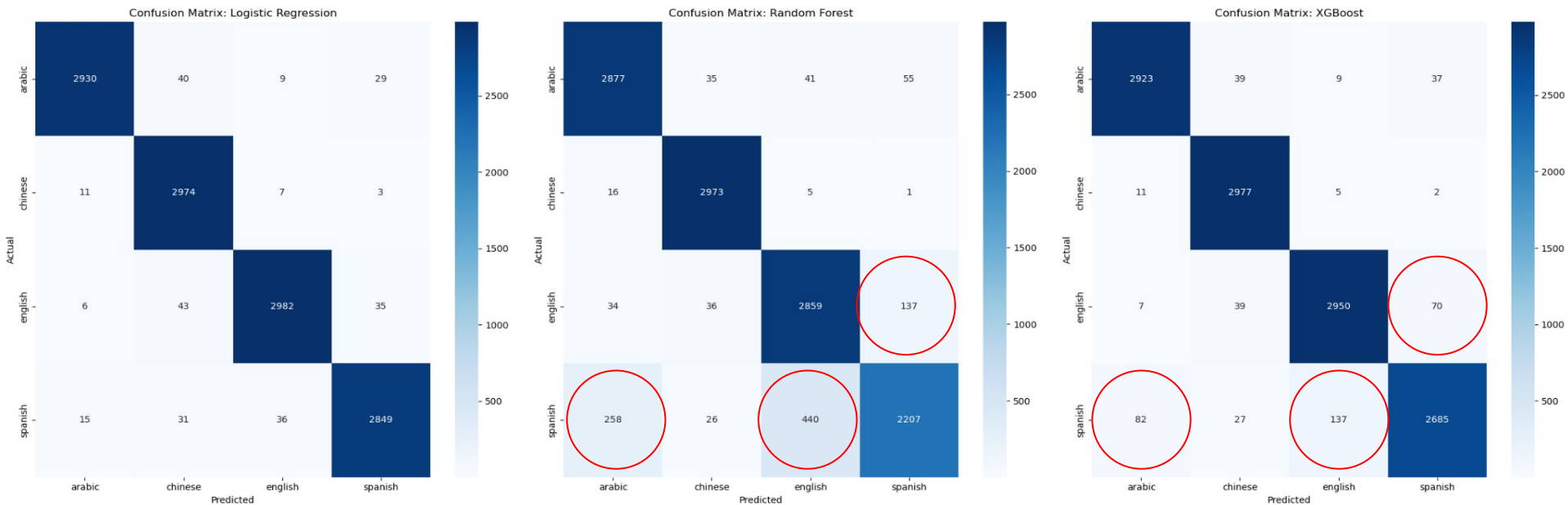
## Random Forest

- Train Accuracy: 99.3%, Test Accuracy: 90.8%
- 91% Precision & 91% Recall

## XGBoost

- Train Accuracy: 99.3, Test Accuracy: 96.0%
- 96% Precision & 96% Recall

# Linear Regression is the winner!

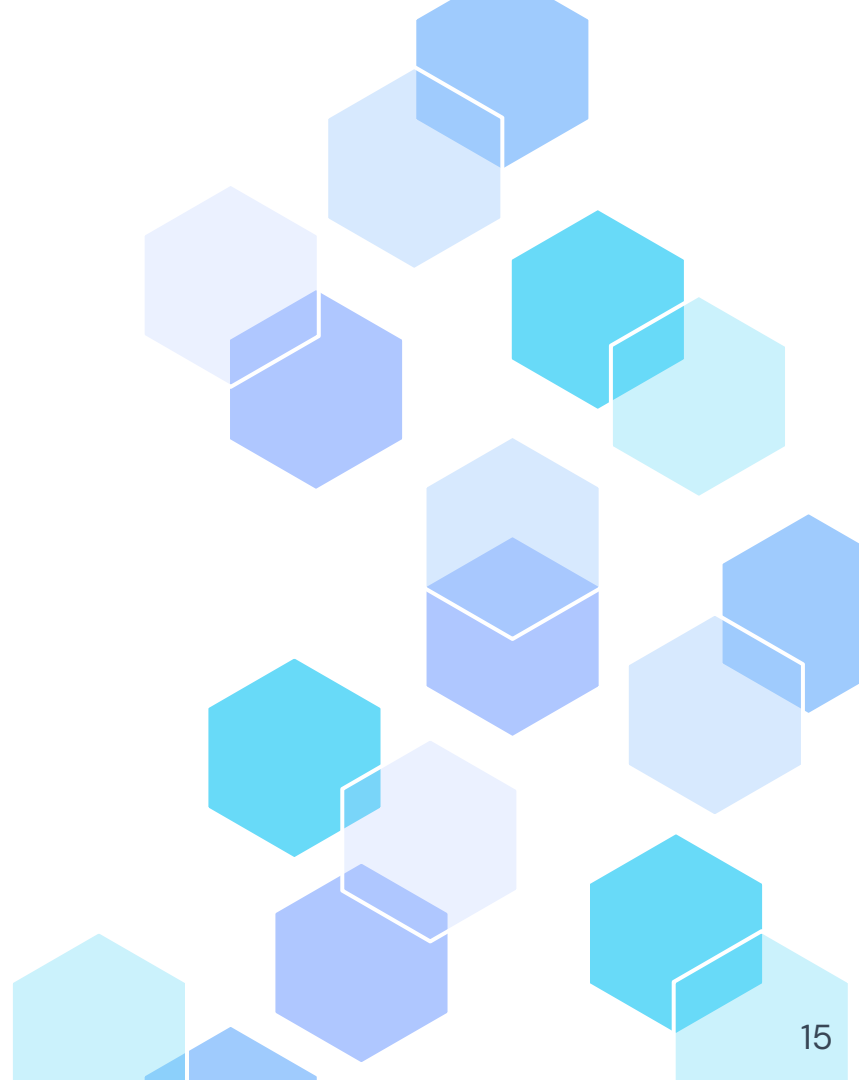


---

# 04

## Adv Models

Generic CNN, VGG CNN Pretrained model



# Generic CNN Model

I used 3 models (Logistic reg, Random Forest, XGBoost)

Observation:

- Training Accuracy: Stabilizes around 32.5%
- Validation Accuracy: Highly variable and unstable
- Overall Accuracy: Both training and validation accuracies are low (30% – 40%)

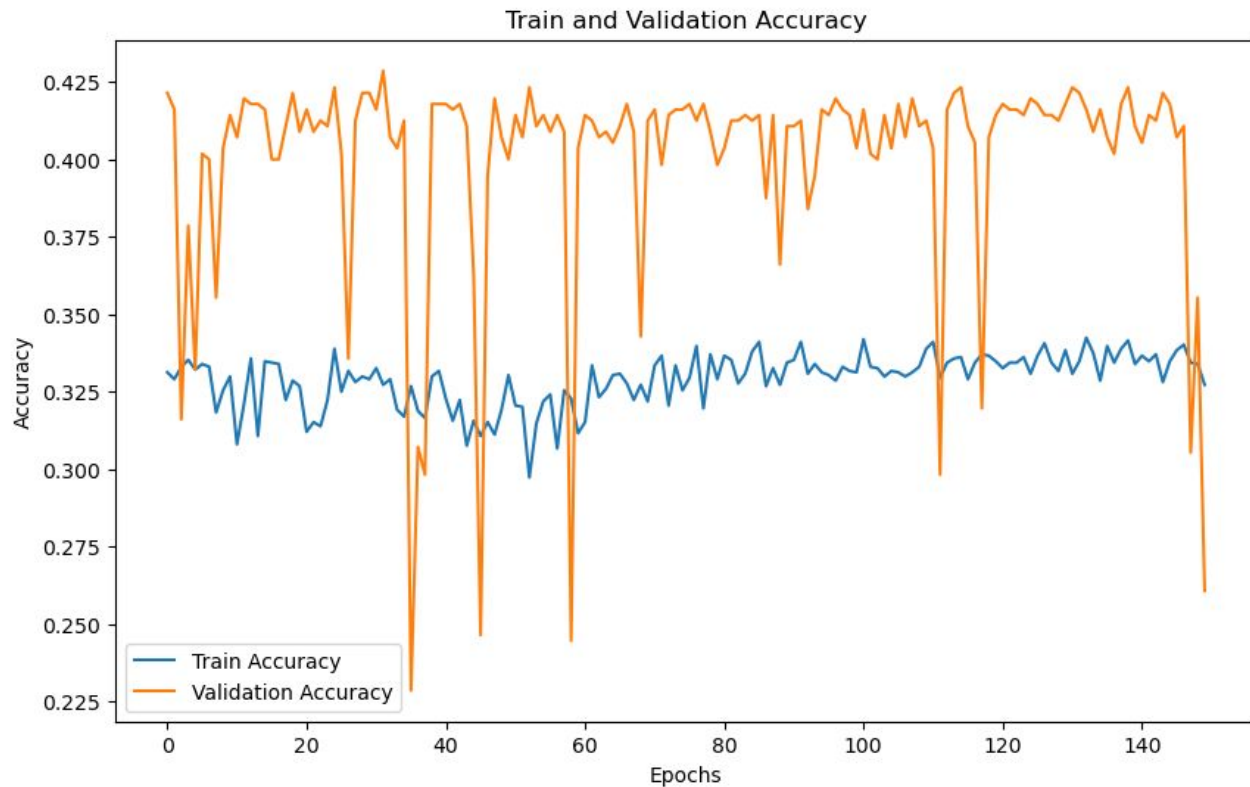
Issues Identified:

- Underfitting: Model may be too simple to capture data patterns





# Generic CNN Model



# VGG CNN Model

## Observation:

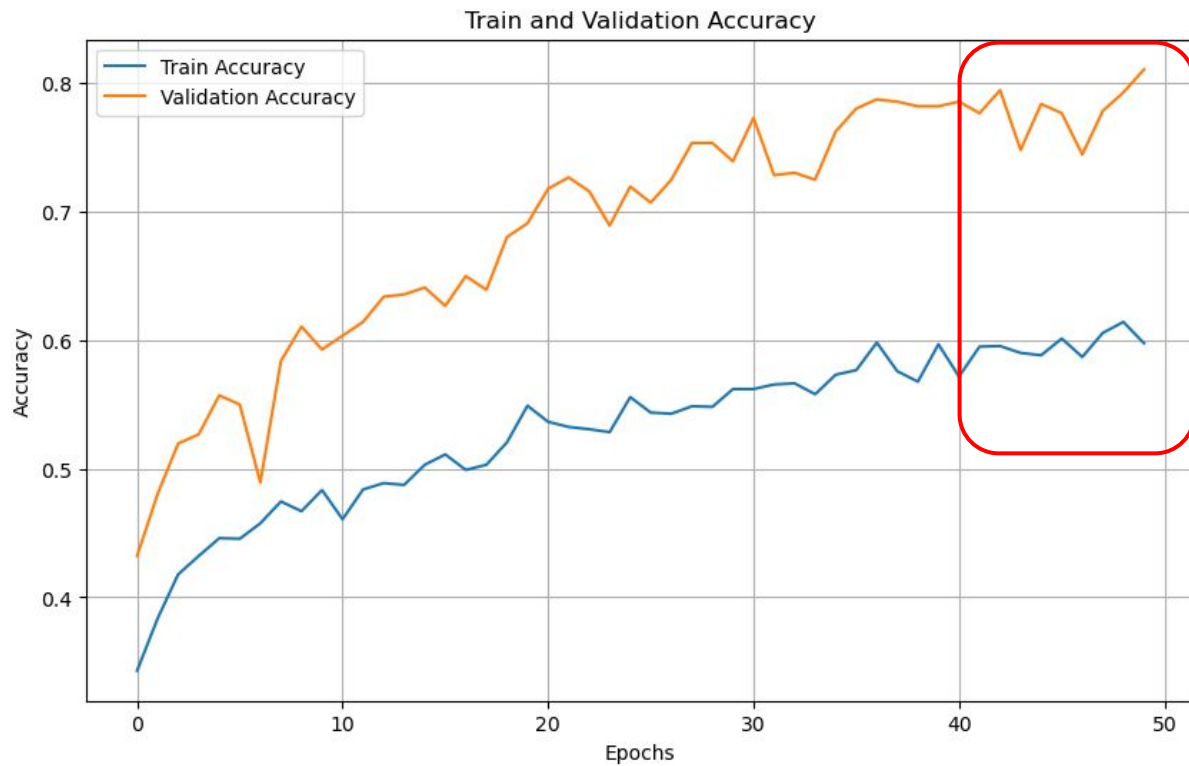
- Training Accuracy: Gradually increases, stabilizing around 60%
- Validation Accuracy: Increases steadily, stabilizing around 80%
- Overall Accuracy: Good generalization

## Positive Indicators:

- Model Learning: Training accuracy improves over time
- Good Generalization: High and stable validation accuracy



# VGG CNN Model

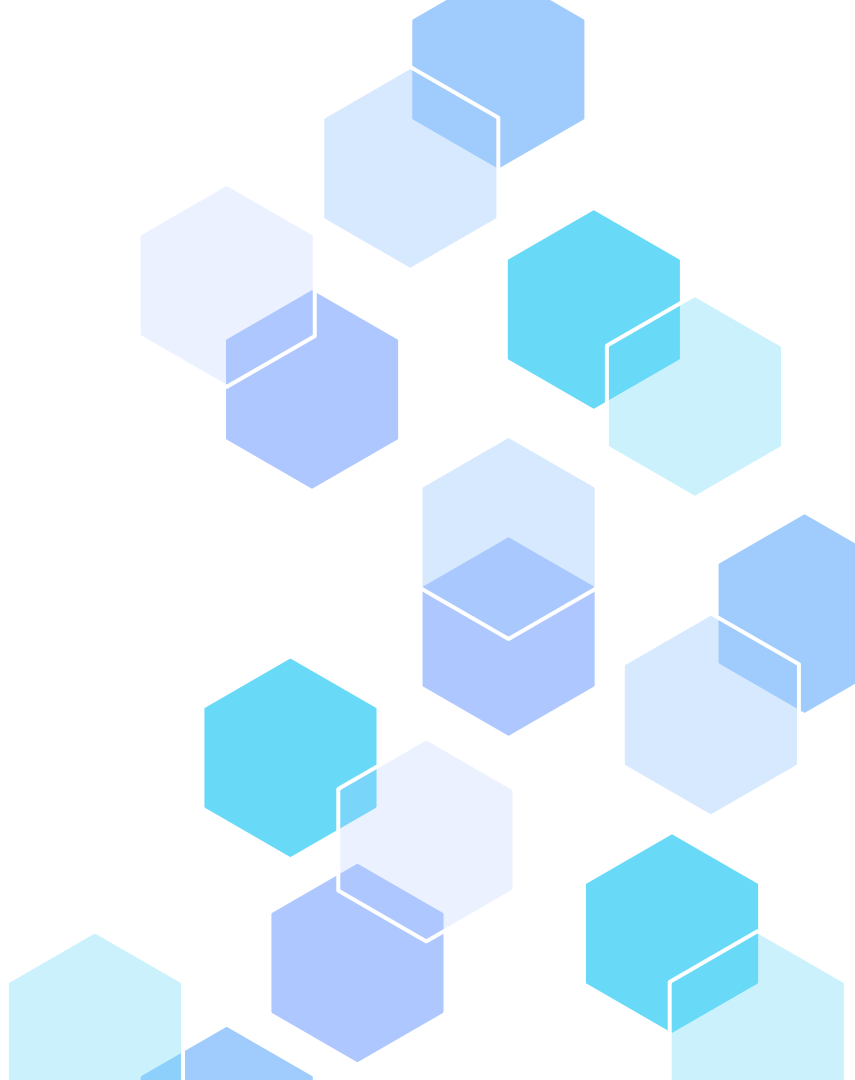


---

# 05

## Next Steps

Advanced Models, Design, Functionality



# Coming Soon...

## Adv Modeling

**CNN:** Further look at other models

**Whisper AI:** Used for transcribing and translation

## Functionality

**Interface:** Create a user friendly front end

**Subtitles:** Customizable subtitle options

## Deployment

**Cloud:** Deploy on the cloud (AWS)

**Sharing:** Release to the public!



---

**Any  
Questions?**