

# Translify - Multimodel Translator With Sign Language Recognition

Hieu Hoang, Abhi Chadha  
Case Western Reserve University

## Introduction

We present Translify, a multimodel translator that breaks down communication barriers by combining spoken language translation with sign language recognition. It enables seamless communication between users of different languages and sign language speakers through a single interface.

## Vision

Our vision is to integrate machine learning and software engineering to leverage users experience with translator app by combining multiple features such as Text-to-Text, Text-to-Speech, Speech-to-Text and Speech-to-Speech, as well as sign language recognitions for people with disabilities.

## Functionalities

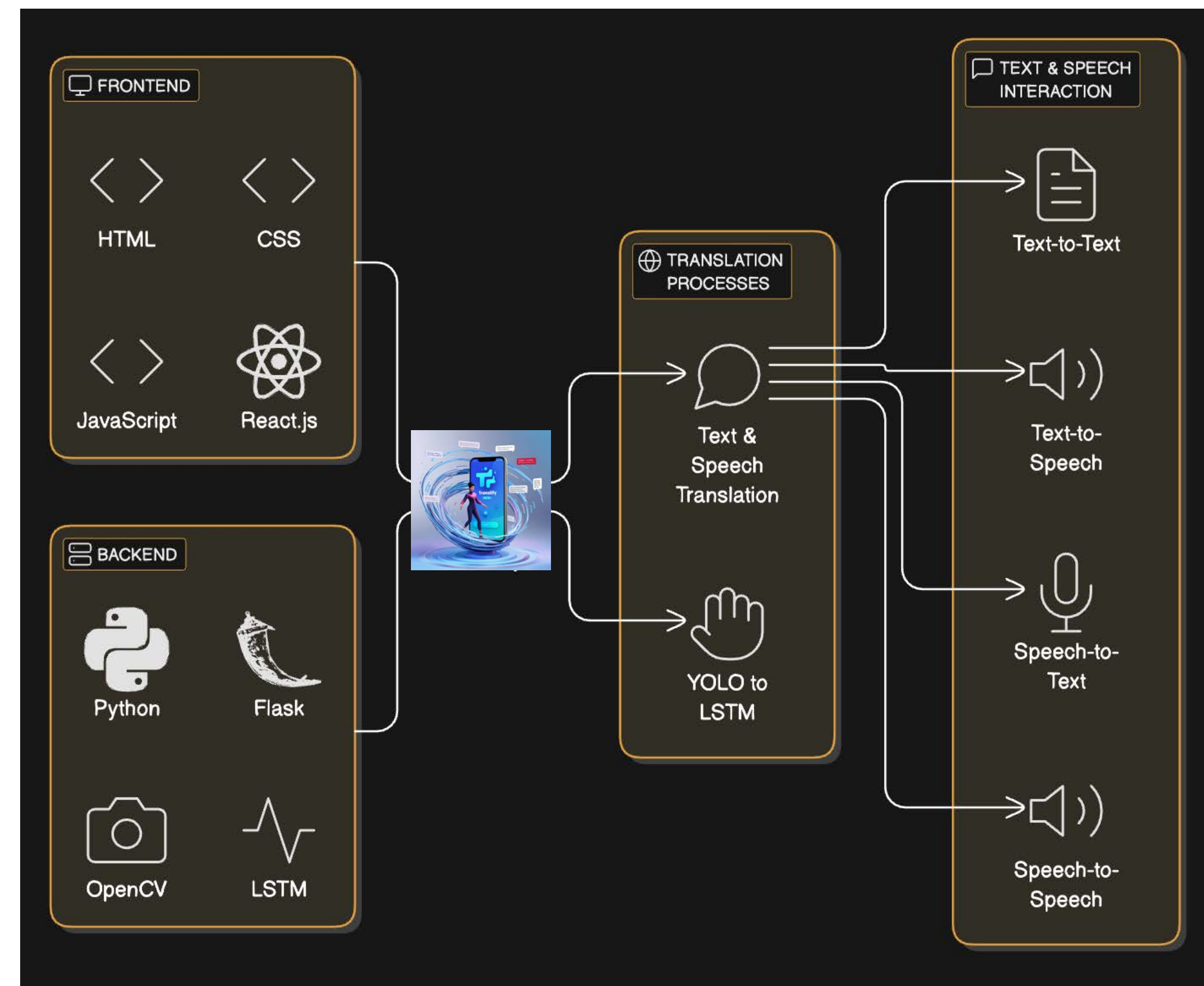
Translify enables real-time translation across five modes: text-to-text, speech-to-speech, text-to-speech, speech-to-text, and sign language recognition. Supporting 106+ languages, it provides a unified interface with dropdown language selection and handles both text and speech inputs seamlessly.

## Language Translation

The system uses specialized APIs with Flask backend, implementing custom caching for frequent phrases and batch processing for multiple requests. Robust database models and security protocols ensure reliable performance and data integrity.

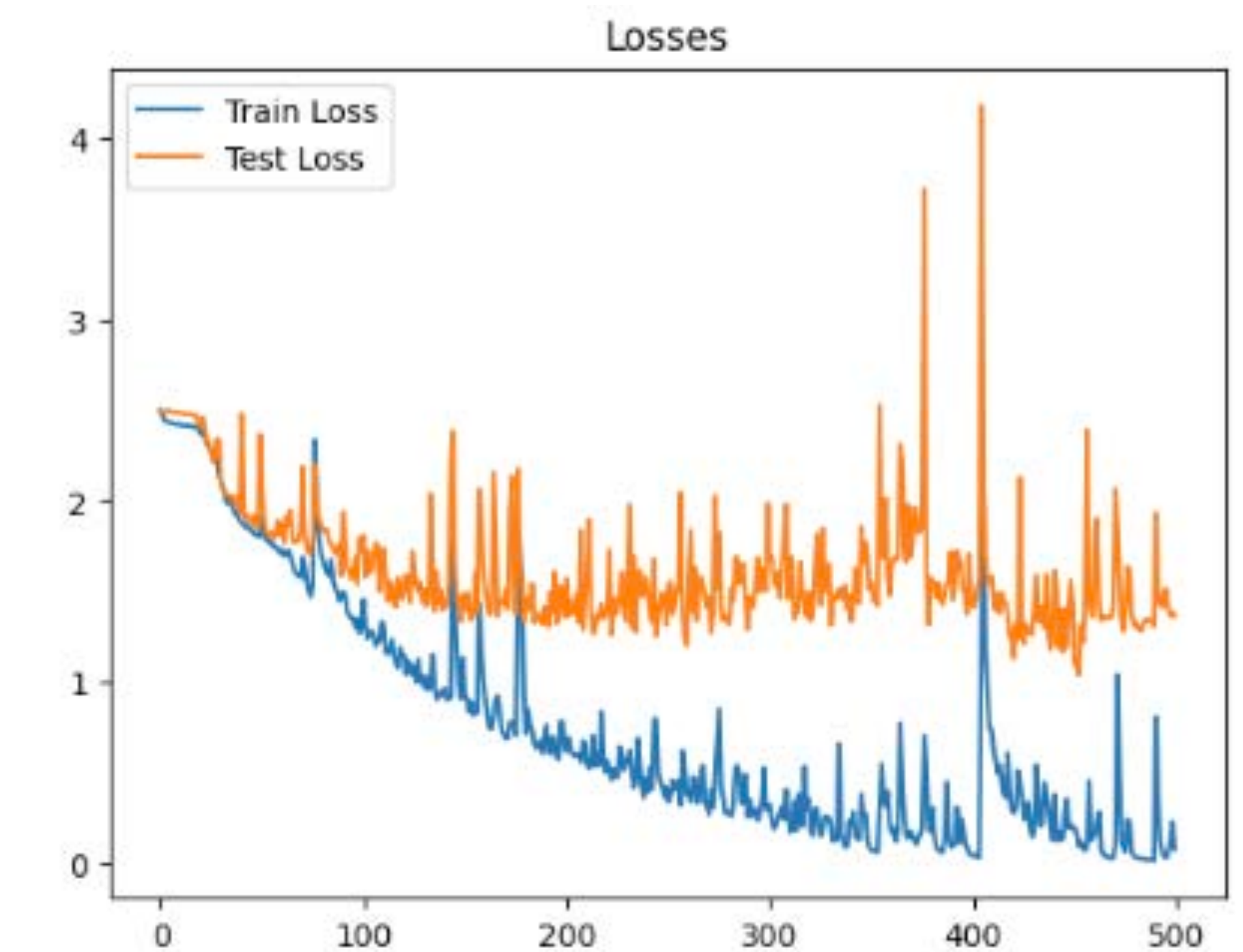
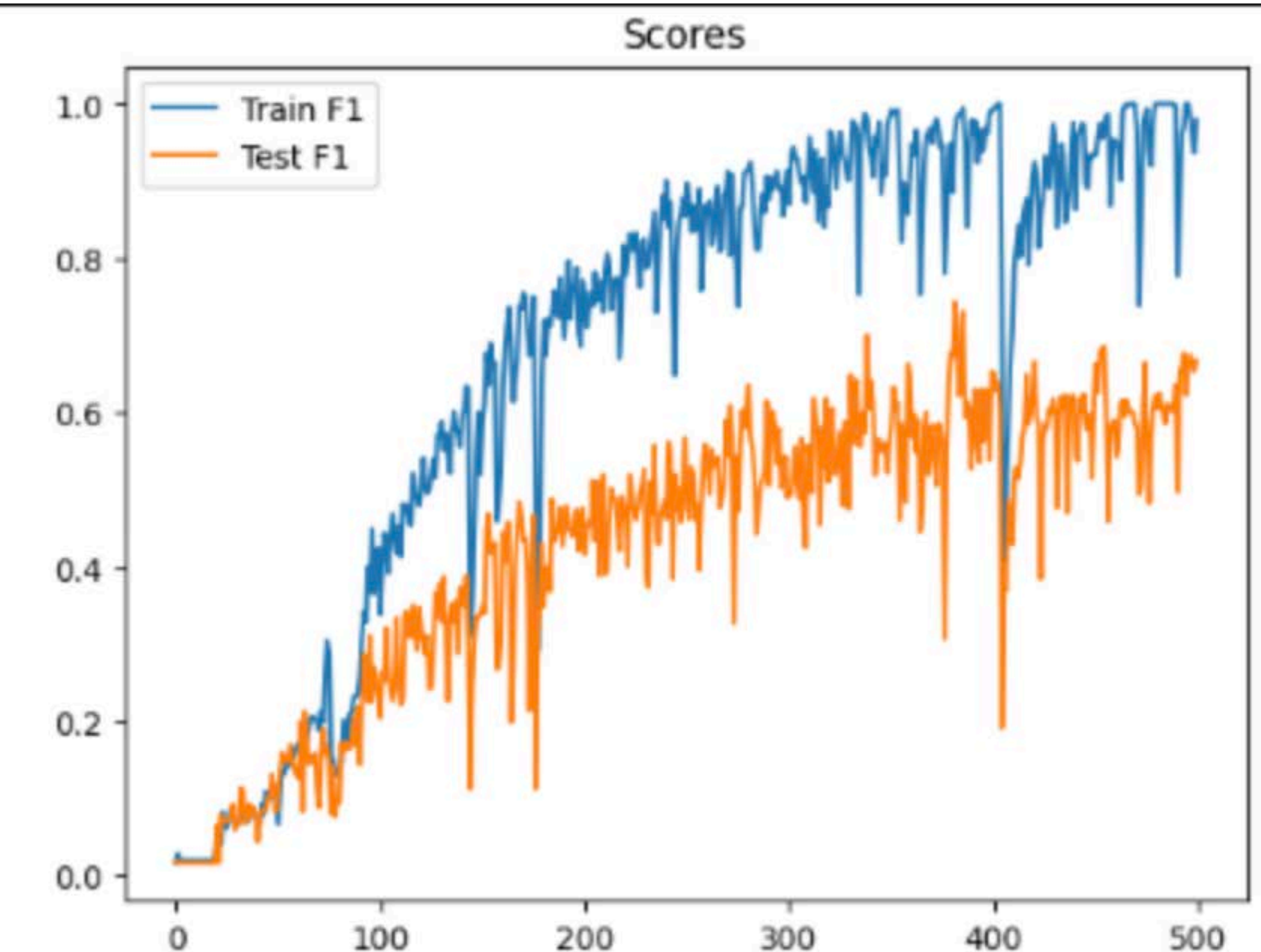
## Sign Language Recognition

Using LSTM architecture and OpenCV, the system processes video at 120 FPS for real-time ASL translation. The model, trained on augmented datasets, creates accurate user-tracking bounding boxes and maintains low latency in gesture classification.



## System Design

- **Frontend stack:** HTML/CSS for markup/styling, JavaScript for DOM manipulation, and React.js for component-based UI development with virtual DOM rendering and state management
- **Backend infrastructure:** Python/Flask RESTful API endpoints, OpenCV for computer vision processing, LSTM neural networks for sequence modeling and prediction tasks
- **Translify architecture:** uses API endpoints to connect React frontend with Flask backend, enabling bidirectional data flow and real-time state synchronization between client and server
- **Core NLP features:** Text-to-text translation using LSTM sequence models, speech synthesis (TTS), speech recognition (STT), and speech-to-speech conversion through YOLO object detection integrated with LSTM for audio processing



## Results