AAI-521 Final Project Team-2

Paul Parks, Bin Lu, Eyoha Girma, Jeremy Cryer

Overview

Problem Statement

- According to google over 466 million people are hearing-impaired with a need to use American Sign Language (ASL).
- However, only ~0.5 Million people actually know how to use ASL.

s

<u>The GOAL</u>

Create a model that can translate ASL to text.

Dataset

Google - Isolated Sign Language Recognition

URL:

- https://www.kaggle.com/competitions/asl-signs/data

Data Contents:

- Multiple Parquet files full of the spatial coordinates of each frame of a raw video.
- A training file matching the parquet files to the sign being performed.

Training File

	path	participant_id	sequence_id	sign	sign_info
0	train_landmark_files/26734/1000035562.parquet	26734	1000035562	blow	25
1	train_landmark_files/28656/1000106739.parquet	28656	1000106739	wait	232
2	train_landmark_files/16069/100015657.parquet	16069	100015657	cloud	48
3	train_landmark_files/25571/1000210073.parquet	25571	1000210073	bird	23
4	train_landmark_files/62590/1000240708.parquet	62590	1000240708	owie	164

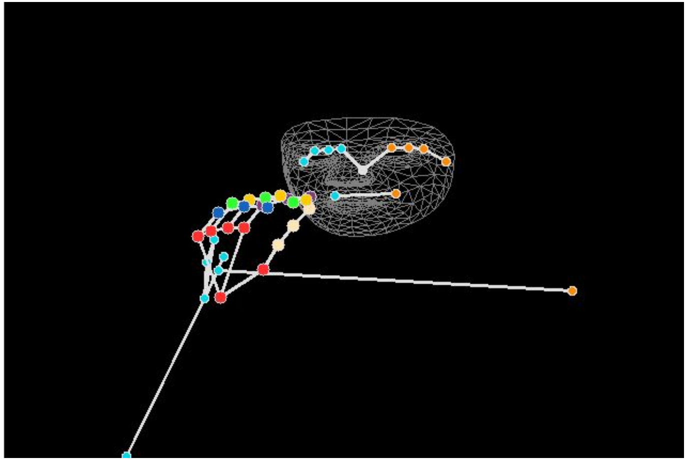
Parquet File

	frame	row_id	type	landmark_index	x	y	z
0	17	17-face-0	face	0	0.495870	0.478694	-0.037412
1	17	17-face-1	face	1	0.492222	0.447209	-0.067939
2	17	17-face-2	face	2	0.492067	0.457237	-0.035722
3	17	17-face-3	face	3	0.480419	0.415996	-0.050779
4	17	17-face-4	face	4	0.492035	0.437453	-0.072314

EDA

EDA

- EDA revealed each Parquet file has landmarks broken down by multiple frames of a raw video of ASL being performed.
- If not inputted correctly, each instance of a frame can be treated as one point turning the example photo into a single point.
- If a hand isn't used for the signing, the data will be null.



	frame	row_id	type	landmark_index	x	y	z
468	20	20-left_hand-0	left_hand	0	NaN	NaN	NaN
469	20	20-left_hand-1	left_hand	1	NaN	NaN	NaN
470	20	20-left_hand-2	left_hand	2	NaN	NaN	NaN
471	20	20-left_hand-3	left_hand	3	NaN	NaN	NaN
472	20	20-left_hand-4	left_hand	4	NaN	NaN	NaN
473	20	20-left_hand-5	left_hand	5	NaN	NaN	NaN
474	20	20-left_hand-6	left_hand	6	NaN	NaN	NaN
475	20	20-left_hand-7	left_hand	7	NaN	NaN	NaN
476	20	20-left_hand-8	left_hand	8	NaN	NaN	NaN
477	20	20-left_hand-9	left_hand	9	NaN	NaN	NaN
478	20	20-left_hand-10	left_hand	10	NaN	NaN	NaN
479	20	20-left_hand-11	left_hand	11	NaN	NaN	NaN
480	20	20-left_hand-12	left_hand	12	NaN	NaN	NaN
481	20	20-left_hand-13	left_hand	13	NaN	NaN	NaN
482	20	20-left_hand-14	left_hand	14	NaN	NaN	NaN
483	20	20-left_hand-15	left_hand	15	NaN	NaN	NaN
484	20	20-left_hand-16	left_hand	16	NaN	NaN	NaN
485	20	20-left_hand-17	left_hand	17	NaN	NaN	NaN
486	20	20-left_hand-18	left_hand	18	NaN	NaN	NaN
487	20	20-left_hand-19	left_hand	19	NaN	NaN	NaN
488	20	20-left_hand-20	left_hand	20	NaN	NaN	NaN

Model - LSTM

Data Pre-Processing

Utilizes **OpenCV** and **MediaPipe** for processing and extracting key landmarks from ASL sign language datasets.

Converts extracted keypoints into numpy arrays, saving them in an organized structure based on sign labels.

Implements a system to identify and use the first **10 consecutive frames** containing consistent left or right hand movements.

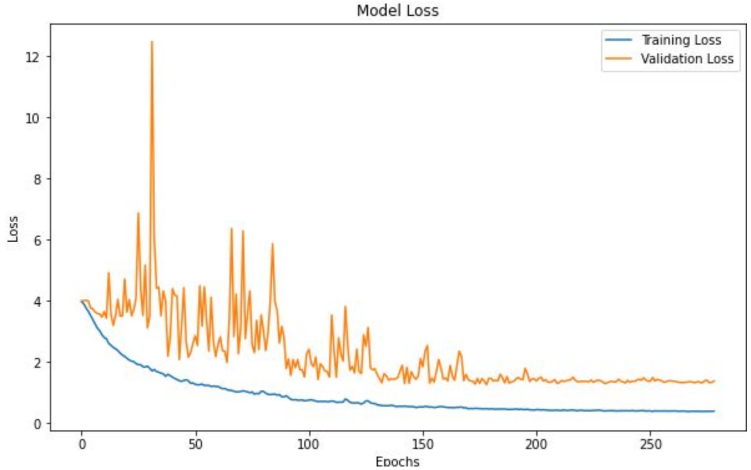
Model LSTM

LSTM model with two layers, the first with **128** neurons and the second with **256** neurons, both using **ReLU** activation.

Incorporates **Dropout** and **BatchNormalization** layers for regularization and to prevent overfitting.

Utilizes a **Dense** layer with softmax activation for multi-class classification of ASL signs.

Employs an **Adam** optimizer with learning rate adjustments and callbacks like **EarlyStopping** and **ReduceLROnPlateau** for efficient training.



Model LSTM - Results

Model Name	Trained Signs	Accuracy	Precision	Recall	F1
All Signs	250	0.48	0.49	0.48	0.48
Removed Low Performing Signs	65	0.76	0.77	0.76	0.76
Top 15	15	0.88	0.89	0.88	0.88

Model - LSTM Bidirectional

Introduction to Advanced LSTM Variants

LSTM and its Importance: Long Short-Term Memory models and their role in sequence data processing.

Project Context: Need for advanced models in ASL recognition.

Focus: Exploring beyond standard LSTM for enhanced performance.

Exploring Advanced LSTM Variants: BiLSTM and GRU

Bidirectional LSTM (BiLSTM) Overview

- BiLSTM Concept: Processes data in both forward and backward directions.
- Contextual Understanding: Captures comprehensive context in sequences.
- Application to ASL: Potential benefits for interpreting sign language.
- Mixed Results: Initial promise but eventual plateau in performance.

Gated Recurrent Unit (GRU) Overview

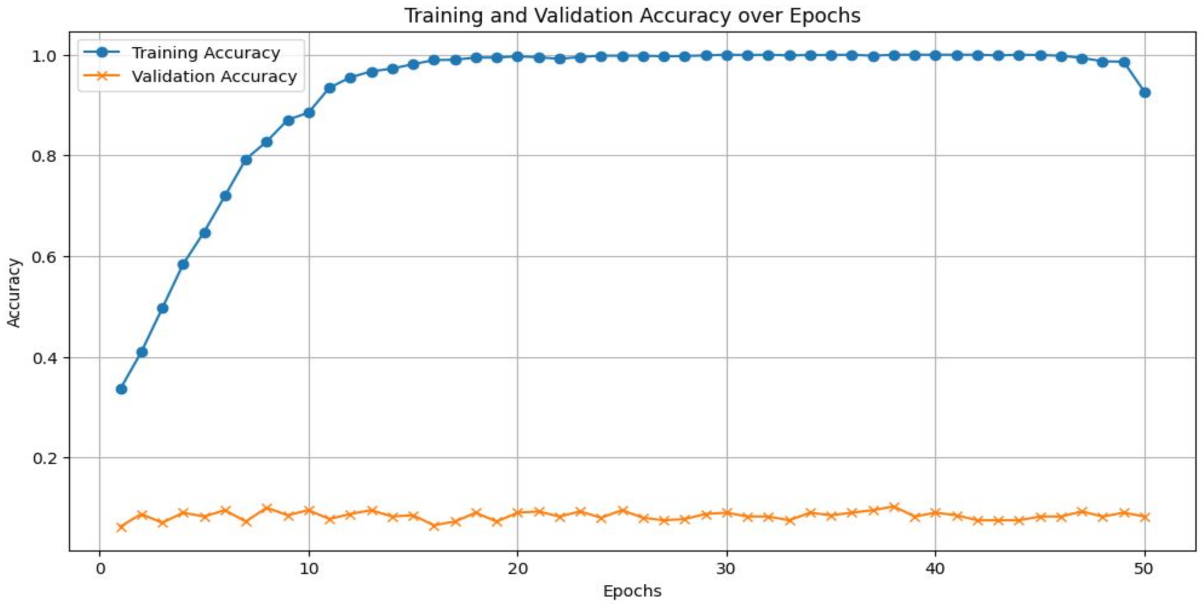
- GRU Mechanics: Simplified version of LSTM with fewer parameters.
- Update Gate: Combines forget and input gates for efficiency.
- Speed and Efficiency: Designed for faster processing.
- Project Application: Considered for ASL data, aiming for streamlined learning.

Challenges with Basic BiLSTM in Our Project

Overfitting Issues: Models too tailored to training data, poor validation performance.

BiLSTM Data Insights: High training accuracy vs. low validation accuracy.

Generalization Struggles: Inability to effectively interpret new, unseen data.



Transition to Transformer Model

Decision Point: Pivot from LSTM variants to a different model architecture.

Transformer Model Consideration: Due to success in sequence modeling tasks.

Transformers Advantage; Excelling in understanding sequences.

- ➢ **Self-Attention Mechanism:** Ability to weigh importance of different parts of input.
- ➢ **Parallel Processing**: Faster and more efficient than sequential processing of LSTMs.
- ➢ **Advanced Feature Capturing**: Better at capturing nuances in complex sequential data like ASL.
- ➢ **Scalability and Flexibility**: More adaptable to different types of sequential data.

Model - Transformer

Data Pre-Processing

Mark Wijkhuizen's Transformer Model Data Pre-Processing:

MediaPipe Parquet File → 3D Tensor (Frame, KeyPoint, Landmark Value)

Frame: 64 Frame

Data File Frame < 64: Padding | Data File Frame > 64: Downsampling

KeyPoint:

MediaPipe Tracking Landmark Keypoints:

Face: 468 + Left Hand: 21 + Right Hand: 21 + Pose: 33 = 542 Landmarks Keypoints

Lip: 40 + Dominant Hand: 21 + Dominant Side Pose (Arm and Shoulder): 5 = **66** Landmarks Keypoints

Landmark Value: [X, Y, Z] value from MediaPipe Tracking: **3**

MediaPipe Parquet File → 3D Tensor **(64, 66, 3)**

Transformer Model

Mark Wijkhuizen’s Custom Transformer Model:

1. Attention Mechanism

   Scaled_Dot_Product function use Softmax layer → selectively ignore and pay less attention to certain part of the input such as padding or irrelevant frames

2. Embedding Layer

   LandmarkEmbedding Class is used to embed the Landmarks.

   Embedding Class is used for positional embedding.

3. Encoder Only

Conclusion

Models Benchmark

Model Name	Trained Signs	Accuracy	Precision	Recall	F1
LSTM	250	0.48	0.49	0.48	0.48
Bidirectional LSTM	250	0.09	0.08	0.08	0.09
Transformer	250	0.71	0.74	0.71	0.71

ASL Gesture Detection

High Overall Accuracy: Transformer model achieved an impressive F1 score of 0.71, showcasing its high accuracy in interpreting ASL.

Exceptional Performance on Certain Signs: Demonstrated superior performance with F1 scores above 0.90 for signs like "airplane," "apple," and "owl."

Balanced Precision and Recall: Maintained a balanced performance with a weighted precision and recall both at 0.71, indicating consistent model reliability.

Scalable and Adaptable Framework: Methodology, utilizing pre processing, MediaPipe, LSTM, and Transformer models, provides a scalable and reusable framework that can be efficiently adapted for various other gesture recognition applications beyond ASL interpretation.

Demonstration

Demo

Contributions

Paul Parks

- EDA, LSTM development, research, training, and testing.

Bin Lu

- EDA, Mark Wijkhuizen's Transformer Model interpretation, implementation, and testing

Eyoha Girma

- EDA, LSTM-Bidirectional development, research, training, and testing

Jeremy Cryer

- EDA, Model Exploration, Report construction and formatting.

Source

- https://github.com/p-parks/AAI-521_FinalProject_Team2