

Sign Speak

Lokeshwar Jha
Pranav Varma
Zhiping Liu
Afia Khan

Team



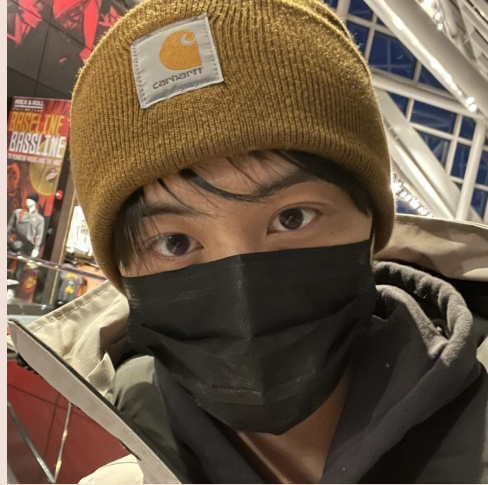
Lokeshwar



Pranav



Afia



Zhiping

Table of content

01

Product
Concept/
Overview

02

Product AI
Canvas

03

Team

04

Value of Data
Calculation for
users

05

Data
Flywheel

06

Model Type
Selection

07

Model
Metrics

08

Inferences

09

Model Results

10

MVP
Development

11

The
Proposed
Architecture

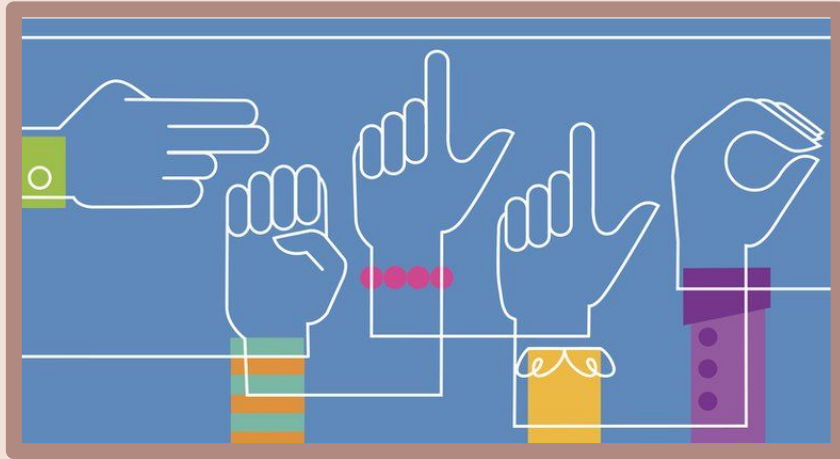
12

End-User
Application

13

Governance
&
Monitoring

Product Concept/Overview



To build an interactive platform through which a mute person can communicate using sign language or text/visuals.

We envision this product to be utilized in customer-care support for specially abled and kiosks in shopping malls.

Product AI Canvas

AI Canvas

Opportunity

Better user experience for customers with hearing or speaking aid

Solution

To read user's gestures through their webcam, process the message and interact with them through text.

Users

specially abled customers such as deaf and mute individuals

Data

MINIST (A-Z signs), ASL (O-9, A-Z signs)
WLASL (Video data)
Scraping

Strategy

Understanding the sign language and replying in text

Success Criteria

Correct response to user

Product Team Roles

Data Engineer/Data Labeller

Data pipeline, Data labelling
Data Versioning

Software Developer

Rest APIs, Model Deployment
User Interaction, User Feedback
Version Control

Machine Learning Engineer

Develop the CNN model
Design the ML pip-line

Product Manager

Make the value analysis
Design the future scope

Value of Data Calculation for User

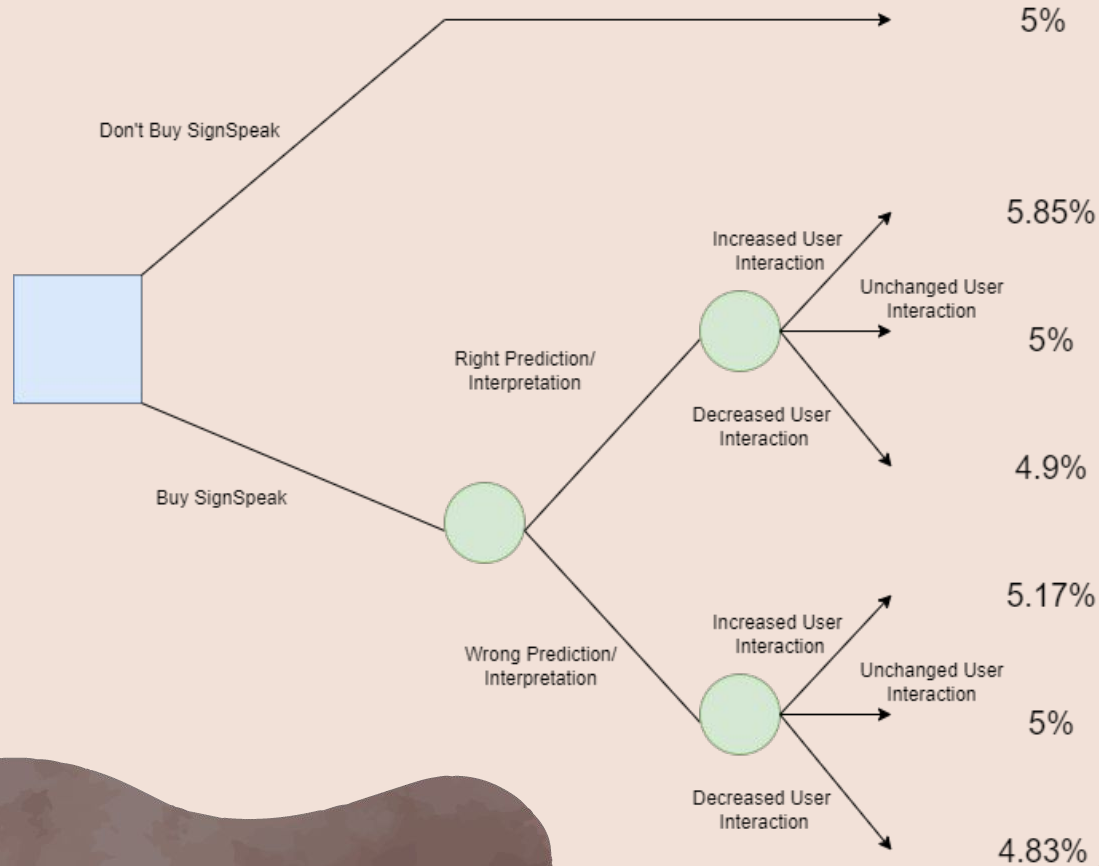
Improving Customer Experience

Texting is one way to interact, it lacks the personalized touch, especially for companies who value customer satisfaction at their core.

This product addresses the needs of the specially abled community to make them feel more welcomed and engaged.



Value of Data Calculation for User



Data Flywheel

Problem
Definition

To convert
hand signs
to texts

Data
Curation

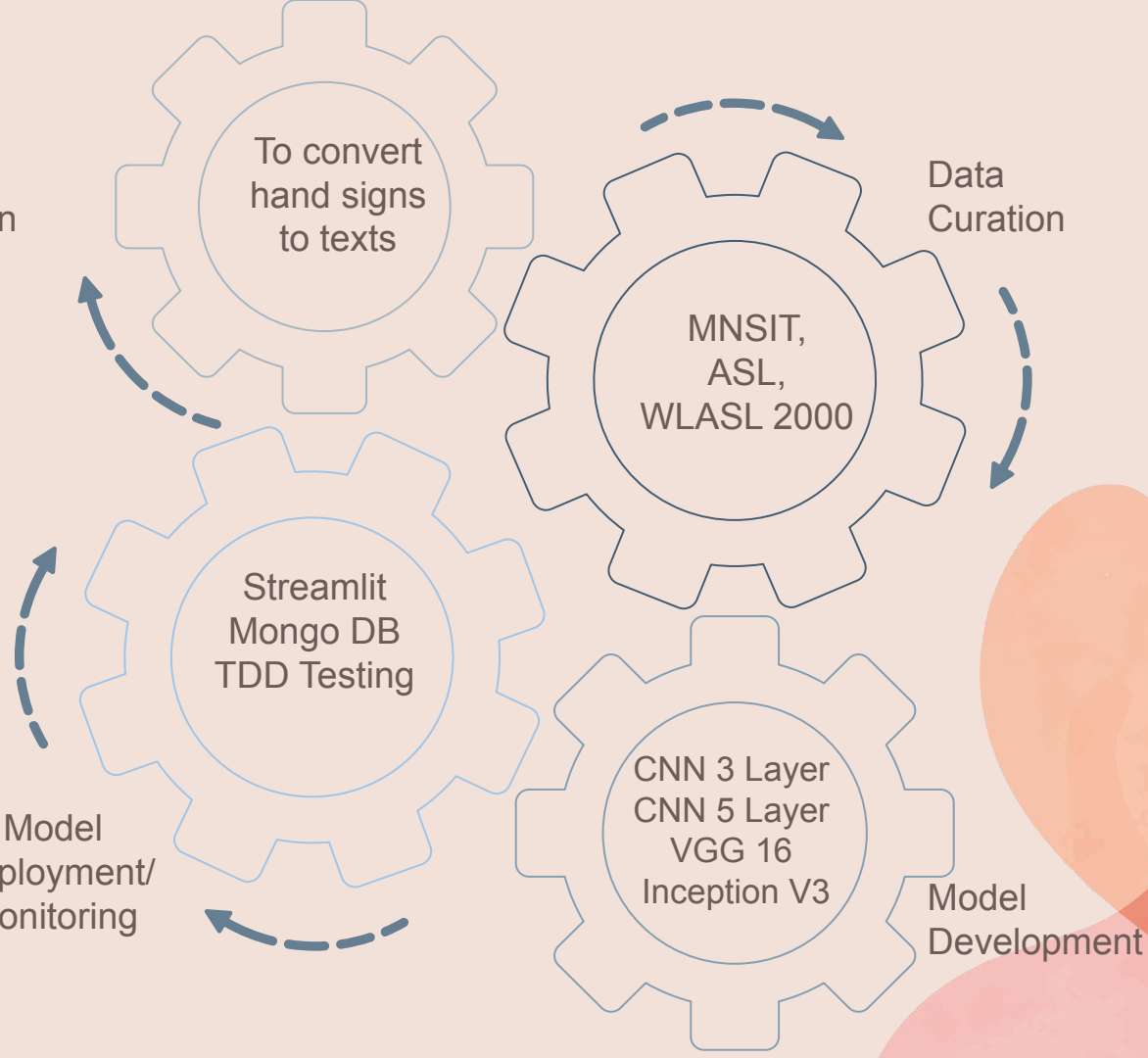
MNSIT,
ASL,
WLASL 2000

Streamlit
Mongo DB
TDD Testing

Model
Deployment/
Monitoring

CNN 3 Layer
CNN 5 Layer
VGG 16
Inception V3

Model
Development



Model Type Selection

Logistic Regression

Baseline Model



CNN 3-Layers,
CNN 5-Layers

CNNs work well with
image classification
tasks



Transfer learning
with VGG-16

Updated the last layer for our
use-case while preserving the
pretrained weights



Transfer learning with InceptionV3

[Inception blocks after convolving with multiple filters seems to give us the best results by far]

Model Metrics

Accuracy

Model Test Accuracy

Precision

Precision on all the symbols, letters & digits

Recall

Recall on all the symbols, letters & digits

Testing in real-time scenarios

[We tried to test the model performances on real-time image captures through streamlit app]

Model Results

Models	Dataset	Accuracy
Logistic Regression	MNIST Dataset (28*28)	Train: 77.5% Test: 74.37%
	Alphabet Images (200*200)	Train: 72% Test: 68.4%
CNN 3 Layers	MNIST Dataset (28*28)	Train: 95.1% Test: 93.2%
	Alphabet Images (200*200)	Train: 98.1% Test: 97.3%
CNN 5 Layers	MNIST Dataset (28*28)	Train: 96.2% Test: 95.3%
	Alphabet Images (200*200)	Train: 98.3% Test: 97.4%
VGG-16	MNIST Dataset (28*28)	Train: 99.8% Test: 99.01%
	Alphabet Images (200*200)	Train: 97.2% Test: 98.5%
InceptionV3	MNIST Dataset (28*28)	Train: 99.9% Test: 99.7%
	Alphabet Images (200*200)	Train: 98.6% Test: 98.9%

Inferences

- Two signs, “M” and “S” are being confused by the model.
- Higher resolution images work well
- Pretrained models seem to work well on the datasets we trained on
- For real-time sign to text conversion, model performance worsens from the experiments
- Various data augmentation would help us in the real-time prediction
- Fingerspelling of larger words and sentences is not very suitable with the current model.

MVP development

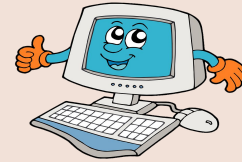
StreamLIT UI App



User Inputs Image of Sign

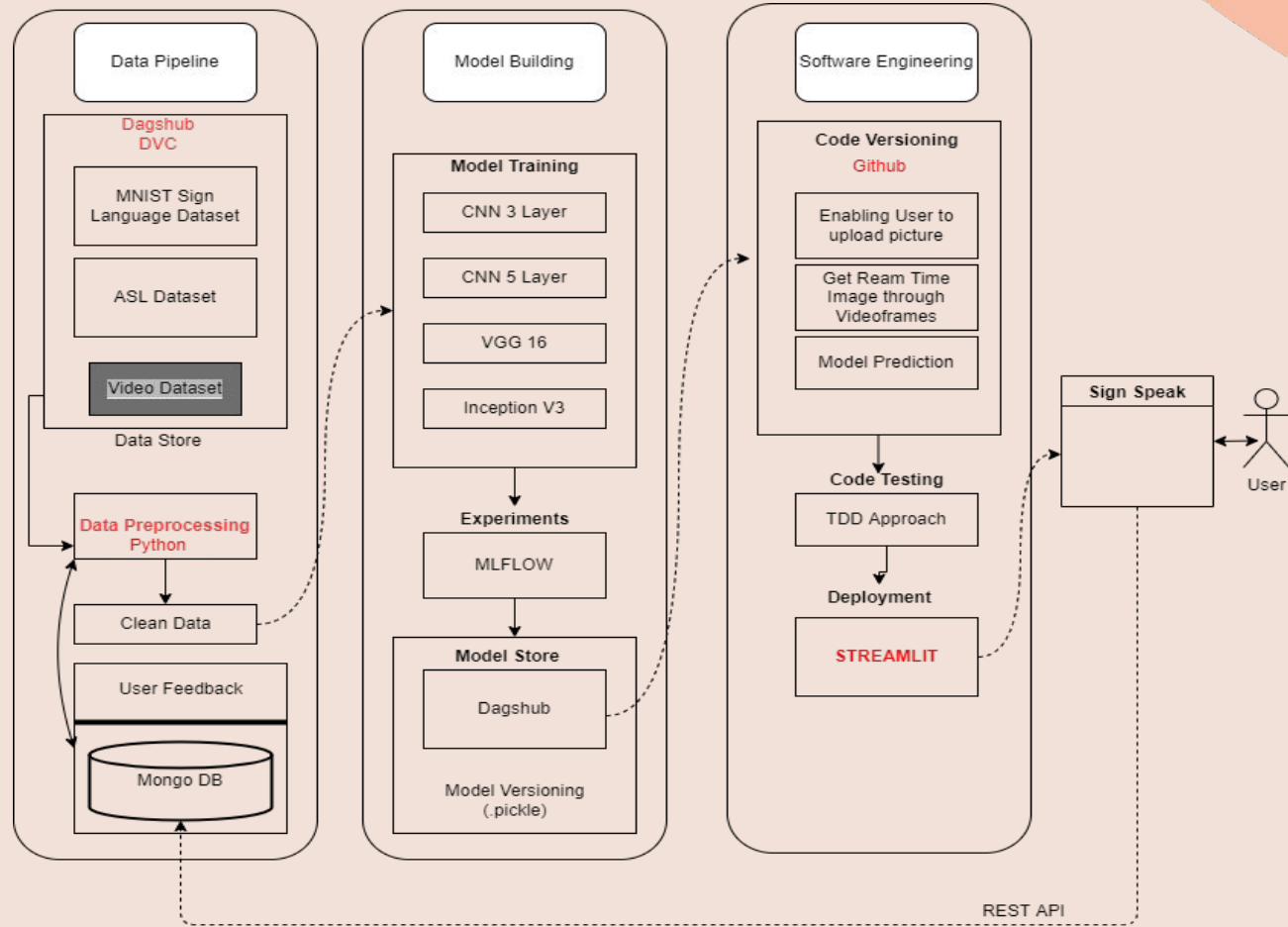


App responds with the meaning/interpretation

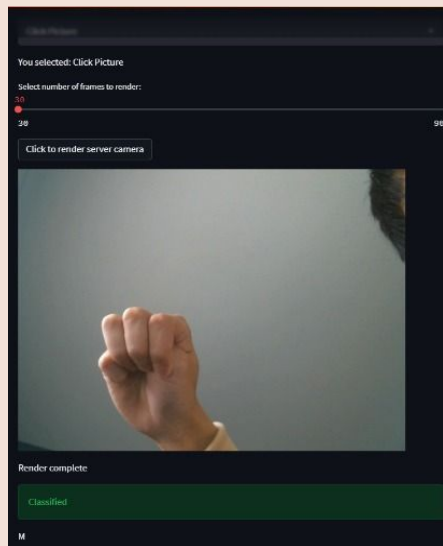


[Product envisioned to be used by customer service orgs/kiosks to better service everyone]

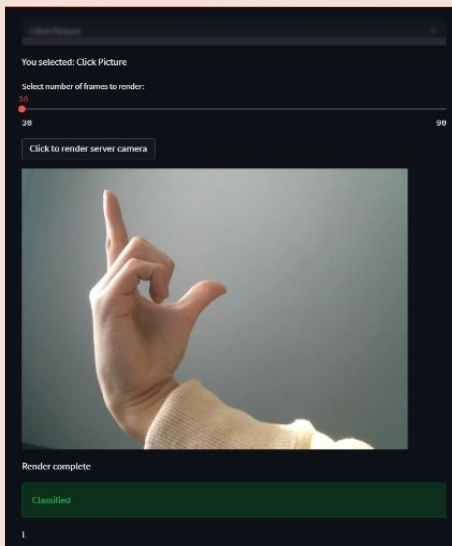
The Proposed Architecture



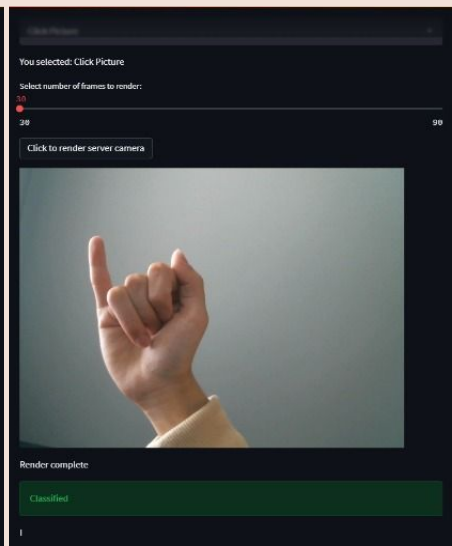
End-user Application



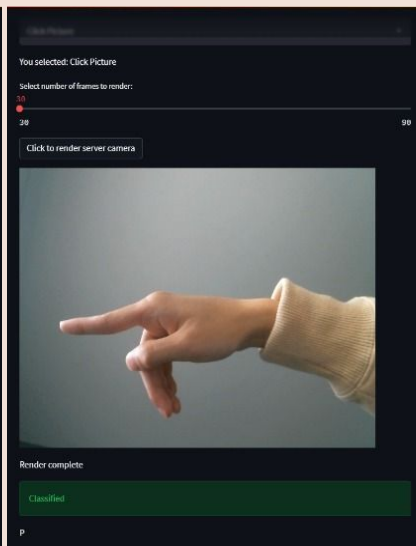
M



L



I



P

Governance & Monitoring

- The dataset that we used is heavily biased on whites.
- Model doesn't seem to work well in other scenarios.
- Planning to scrape balanced data to further improve the model
- Sign languages are often region based. In order to make it more robust, can train for other languages
- Monitoring model and data drift through the user feedback in order to improve model performance.

Project Repo

- Dagshub: <https://dagshub.com/loki2124/SignSpeak>
- Streamlit: <https://share.streamlit.io/loki2124/signspeak/main/app.py>

THANK YOU!

