**Fingerspelling Interpretation**

**Arunima Ayshee**

[**160204066@aust.edu**](mailto:160204066@aust.edu)

**Tahira Salwa Rabbi Nishat**

[**160204070@aust.edu**](mailto:160204070@aust.edu)

**Zannatul Ferdous**

[**13.02.04.070@aust.edu**](file:///I:\4.2%20final%20semester\Soft%20Computing\assignments\project\13.02.04.070@aust.edu)

1. **Introduction:**

Fingerspelling is a subset of sign language used for communication by representing a word or other expression by rendering its written form letter by letter in a manual alphabet. It is a method of spelling words using hand movements.

In this project, we have worked with 26 different letters and 3 different symbols of daily use of the English language. We are making a system that can recognize hand symbols and give us the corresponding meaning as output. A graphic illustration of American Sign Language is shown in below figure -



Fig 01: American Sign Language

1. **Motivation:**
2. Fingerspelling is a survival tool for deaf people when it comes to learning, communicating with rest of the world. It bridges the gap between the learner’s language and sign language. Some deaf people like to use fingerspelling more than others. When we are naming something like a book, people, movie, brands, and places it is easier to fingerspell because there is not always a symbol for the proper name we are trying to fingerspell. For instance, we cannot say "K- Mart" without spelling it out. The idea of fingerspelling is to give us another method in sign language to help the betterment of understanding all around, especially with names that don't really have signs.
3. Some people we come to meet did not grow up learning sign language, instead used a coding system like SEE or the Rochester method (full fingerspelling of English sentences). Also, the community of deaf people comes from a variety of backgrounds, and each background has varying levels of fingerspelling. That’s why it’s an important skill to master so that we can understand deaf individuals when we come in contact with them. Moreover, it also allows us to learn from deaf people and helps to understand the unknown meaning of signs used by them during a conversation.
4. We had to work with RGB images with high resolution. So we had to convert it into grayscale and Transform them into what we can feed to network. Also, the length of the dataset was huge. So the code took a long time to run.
5. **Related Works**:
   1. Object detection, detecting ASL using webcam, real time ASL testing.
   2. Most of the works have been done on ASL till now is based on Convolutional Neural Network, other real time ASL detection work, they have used YOLO which is an object detection neural network based on the CSP approach, Some work were done using Resnet.
   3. We have used shallow Neural Network for our project, it simplified the task maintaining a good accuracy.
6. **Project Objective(s):**
   1. **Tasks of the system:**
      1. Data Preprocessing: We have taken our dataset from Kaggle ASL dataset.

We had to add label for each class of data. Then we calculated how many images of each class we have in our training data. All of the classes have got equal training data.

* + 1. We processed the dataset, transform the images into 224\*224 sized grayscale images using torchvision.transforms .
    2. Then we made our dataset using Imagefolder which returned the images and labels.
    3. We split the dataset into train:validation(80:20) set.
    4. We used Dataloader using this dataset.
    5. Then, we created our model NeuralNetworkModel(), trained it with train data set, and tested it with the validation set.
    6. Finally we tested our model and evaluated its performance.

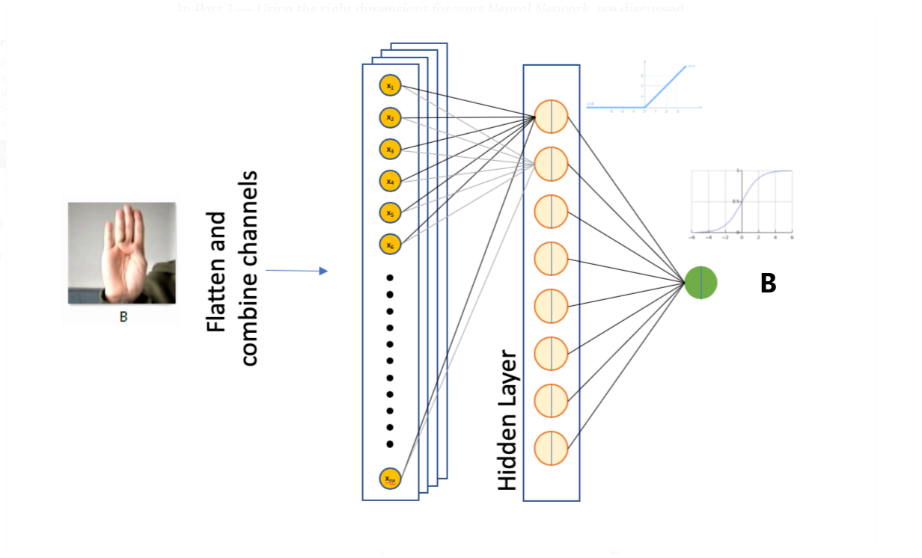


Fig 02: Neural Network Model

* 1. **Dummy Input:**



**Dummy Output:**



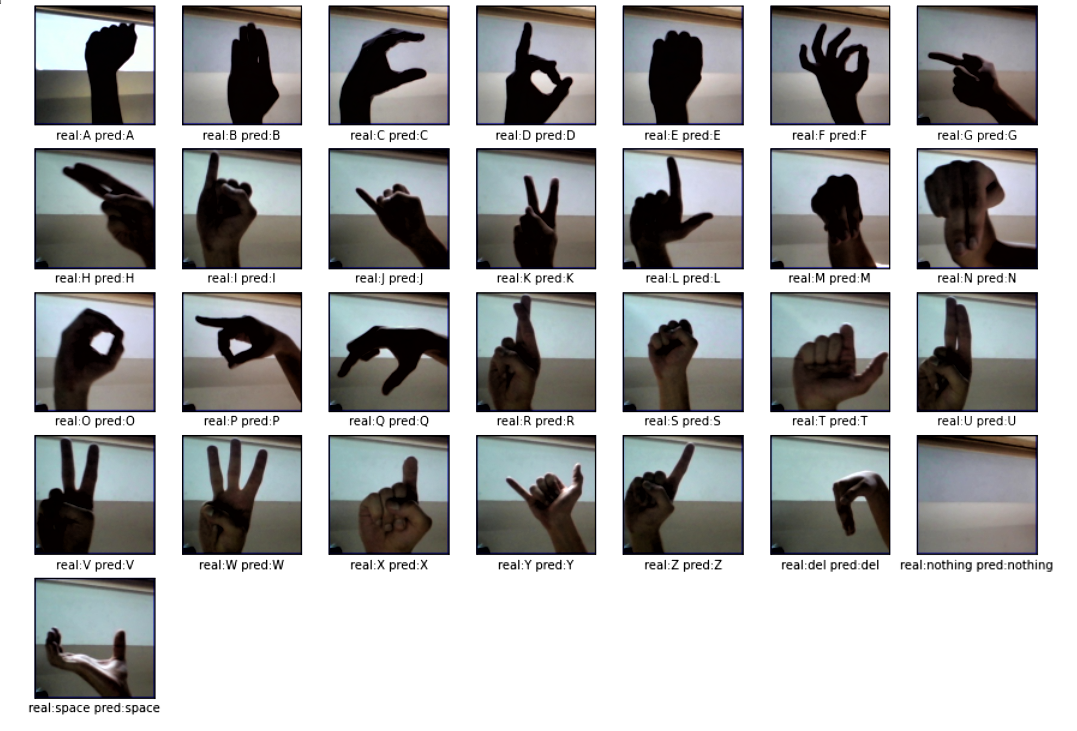
1. **Methodologies / Model**
   1. We have discussed this part in section 3
2. **Experiments**
3. **Dataset**
   * 1. Our Dataset is ASL dataset from Kaggle:

[kaggle datasets download -d grassknoted/asl-alphabet](file:///I:\4.2%20final%20semester\Soft%20Computing\assignments\project\kaggle%20datasets%20download%20-d%20grassknoted\asl-alphabet)

The total number of images in our dataset is 87,000.

We Have 29 classes from A-Z, and del, nothing, space. Each class contains 3,000 images.

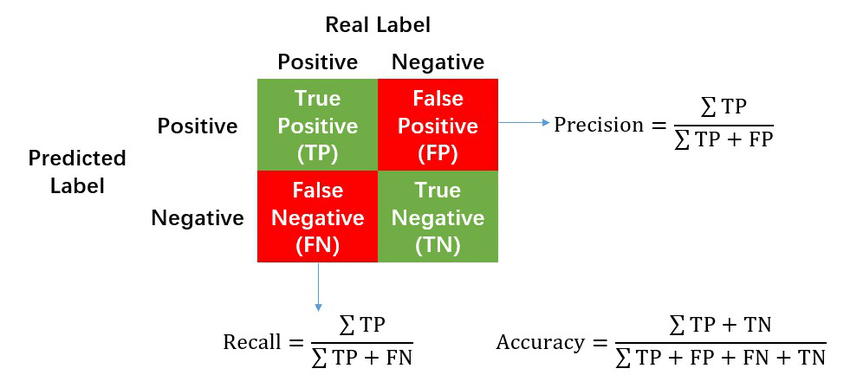
* + 1. Here is a sample of our dataset where we have showed one first picture from each classes.



**Fig 03: Result of test data**

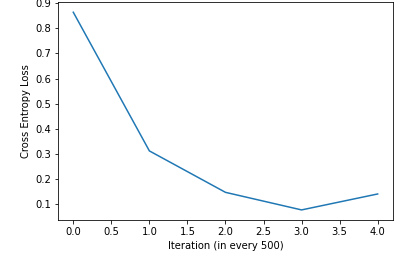
* + 1. Our total dataset from Kaggle had two directory, train directory and test directory. We split the train data from train directory into train: validation = 80:20, so the length of train dataset was 69600 and validation dataset was 17400. The Test directory from Kaggle 28 test images, but we need at least 29 images, one from each classes to evaluate our model. So, we made our own test directory, it had 29 images one from each class.

1. **Evaluation Metric**
   * 1. We evaluated our model on the basis of accuracy. Then we made the evaluation matrix and calculated precision, recall, f1-score done on validation set. We have measured the accuracy for our test dataset too.



**Fig 04: Confusion matrix**

1. **Results** 
   * 1. Here are our best 3 results we have gotten so far:
     2. We have achieved 97.29885057471265% accuracy and the cross entropy loss graph looks like this:



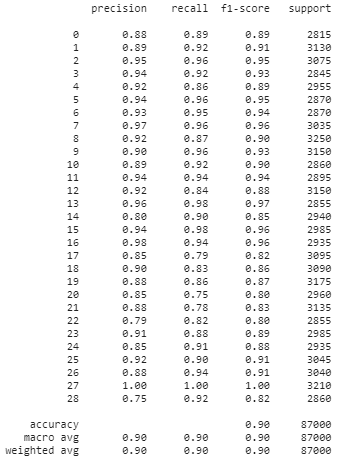
**Fig 05: Cross Entropy loss in each 500 iteration graph (1)**

It looks a little bit overfitted. But all of our test data were classified correctly.



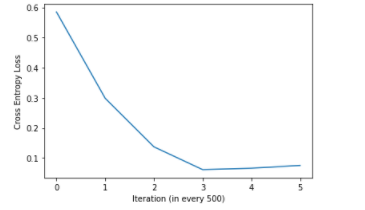
**Fig 06: Result of test data (1)**

The evaluation matrix looked good too.



**Fig 07: Model Evaluation 1**

* + - 1. We have achieved 98.79885057471265% accuracy and the cross entropy loss graph looks like this:



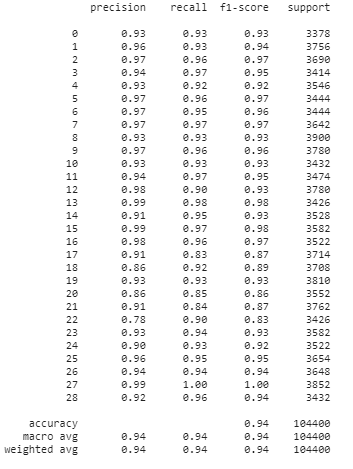
**Fig 08: Cross Entropy loss in each 500 iteration graph (2)**

It looked better. And all of our test data were classified correctly.



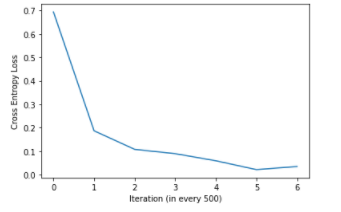
**Fig 09: Result of test data (2)**

The evaluation matrix looked better too –



**Fig 10: Model Evaluation 2**

* + - 1. We have achieved 98.95402298850574% accuracy and the cross entropy loss graph looks like this:



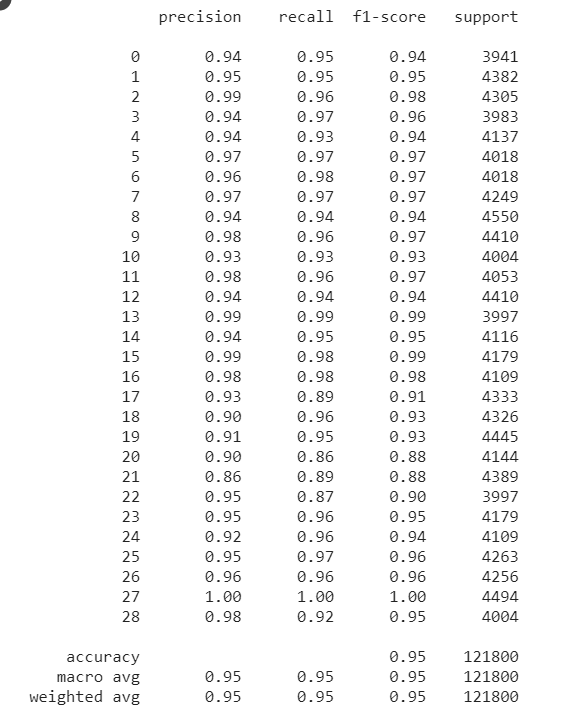
**Fig 11: Cross Entropy loss in each 500 iteration graph (3)**

It looked nice. And all of our test data were classified correctly.

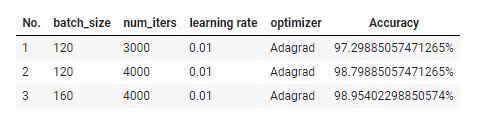


**Fig 12: Result of test data (3)**

This is how our model evaluation was for this case.



**Fig 13: Model Evaluation 3**

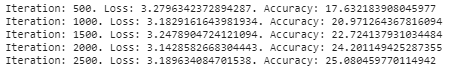


**Fig 14: Hyperparameters for best 3 settings**

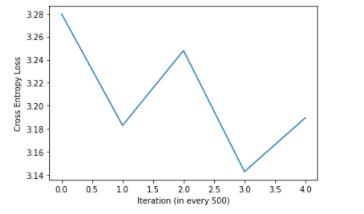
* + 1. Our dataset was big. So, we took batch size between 100-160. For learning rate we took either 0.001 or 0.01, but 0.01 gave better result. Hidden layer was 200.

First**,** we tried with deep neural network

Linear-> Tanh-> Linear-> LeakuRelu-> Linear-> Relu6-> Linear-> Sigmoid with and Adamax optimizer and cross-entropy loss function. The result was poor. Nearly 30%.



Also the cross-entropy loss graph looks bad:



**Fig 15: Cross Entropy loss in each 500 iteration graph for trial settings.**

We used this combination in our model with various

hyperparameter but result didn’t change significantly.

So, we changed the model and used four layer Linear network for

fc1 -> fc2 -> fc3 -> fc4 while varying hidden layer dimension for each layer 224\*224 (= 50176) -> 512 -> 256 -> 128-> 29 and forwarded them using ReLU. We also changed our optimizer to add a graph. The accuracy spiked over 90%. We later tuned more and got 98% accuracy.

1. **Conclusion:**

This system is designed specifically for deaf and hard of hearing people/students. With the help of the fingerspelling method, one can spell the letters of a word in order to get to the meaning. When a person learns sign language, the first thing he learns is the alphabet, the alphabet is going to be signed with the fingers. And once he had learned the alphabet then he will move on to fingerspelling. But a hard situation may arise if they come to appear before a person who can speak and/or without any knowledge of fingerspelling. Our system wants to help those people and make it easier by letting them express their words to the world.

**References:**

* + - 1. <https://en.wikipedia.org/wiki/Fingerspelling>
      2. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3480647/>
      3. <https://www.nidcd.nih.gov/health/american-sign-language>
      4. <https://www.verywellhealth.com/jobs-using-sign-language-1046849>
      5. https://ieeexplore.ieee.org/abstract/document/7033173 - American Sign Language Recognition Using Leap Motion Sensor
      6. https://ieeexplore.ieee.org/abstract/document/7424365 - Real-Time American Sign Language Recognition System Using Surface EMG Signal