SIGN LANGUAGE ALPHABET RECOGNITION USING ANN AND CNN

Submitted By:

Janvi Arora

2K18/SE/070

Mayand Kumar

2K18/SE/078

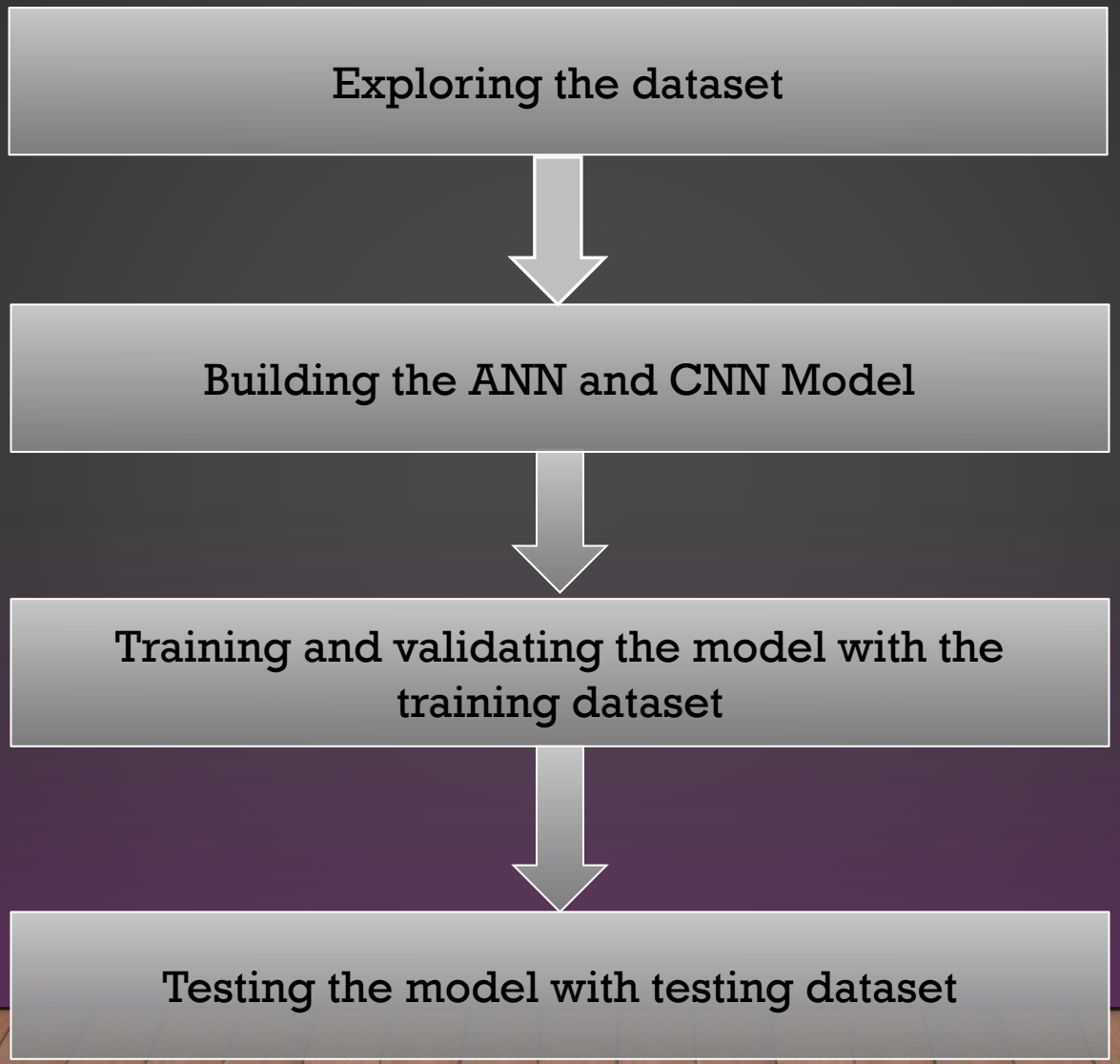
Submitted To:

Dr. Anurag Goel

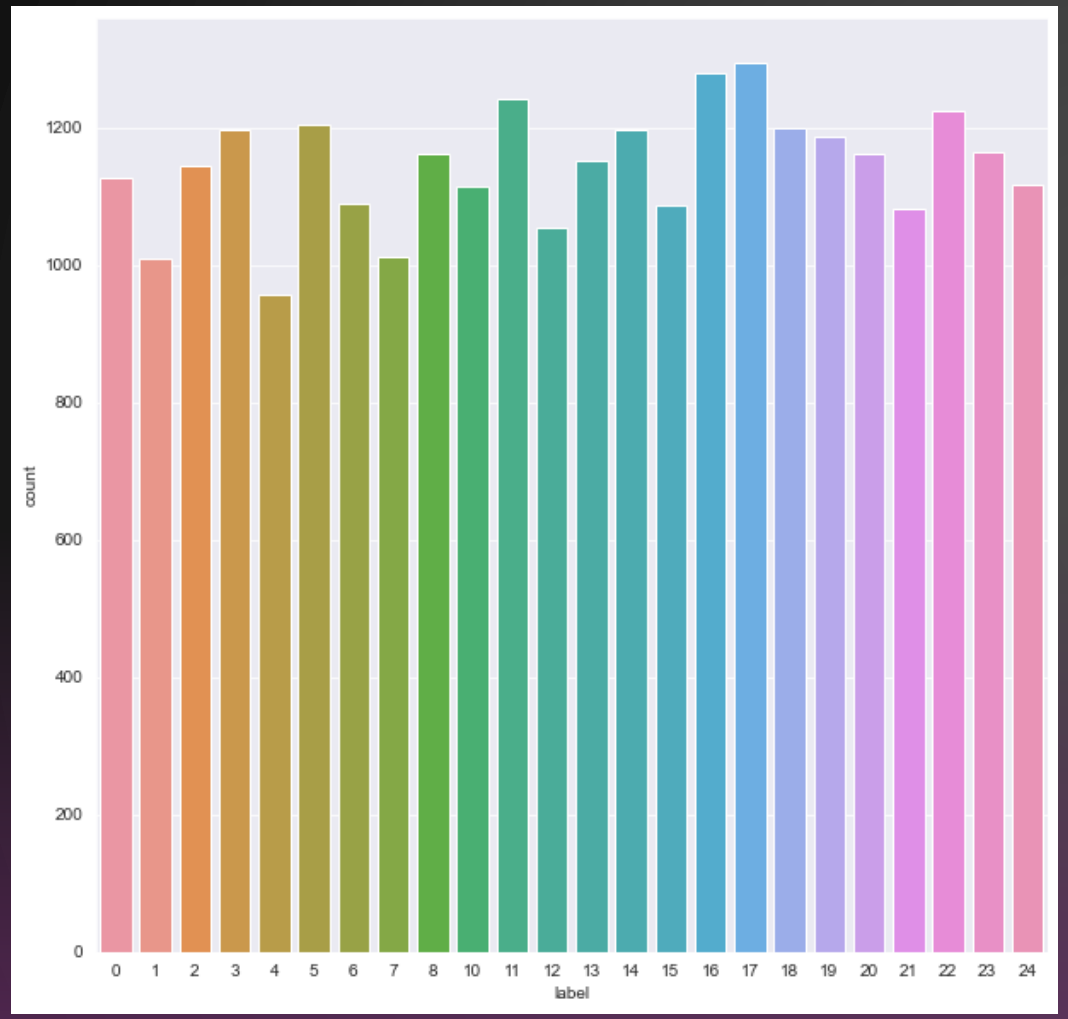
INTRODUCTION

- The sign language (SL) is made by specifications of hand and facial idioms to express their views and thoughts of speech and hearing disabled persons with the normal (speech and hearing) people. Most normal persons may not clearly understand sign language. Therefore, there is a massive communication gap between deaf communities with the general public.
- ASL is the most widely used SL in the world and fourth most usable linguistic in North America.
- ASL includes a set of 26 gesture signs known as an American Manual Alphabet that can be used to spell out many of the English words available. The 19 various hand shapes of ASL are cast-off to make 26 American Manual Alphabets. The recognition of sign language will make communication easier if one of the person isn’t well versed with the language.

STEPS FOR MODEL CONSTRUCTION AND VALIDATION



DATASET DESCRIPTION AND EXPLORATION



- The dataset format is patterned to match closely with the classic MNIST. Each training and test case marks a label from 0 to 25 as a matched map for every alphabetic letter A-Z (and no cases for 25=Z due to gesture motions). The training data (27,455 samples) and test data (7172 samples) are approximately half the size of the standard MNIST but otherwise similar with a header row of the label, pixel1, pixel2…. pixel784 which represents a single 28x28 pixel image with grayscale values ranging 0-255. The original hand gesture image data represented multiple users repeating the gesture against entirely different backgrounds. The Sign Language MNIST data came from greatly extending the tiny variety (1704) of the colour images enclosed as not cropped around the desired hand region .

- The dataset seems balanced as for each training label, and enough training examples exist shown in Figure on the left side that we obtained during the exploration of the dataset.

PERFORMANCE MEASURES AND STATISTICAL METHOD USED

Performance Measures used:

- Accuracy
- Precision
- Recall
- F1 score
- AUC Score

Statistical method Used:

- ReLU Activation Function for input and hidden layers
- Softmax Activation Function for Output layer.
- adam optimizer with learning rate of 0.001 and B1=0.9 and B2=0.999

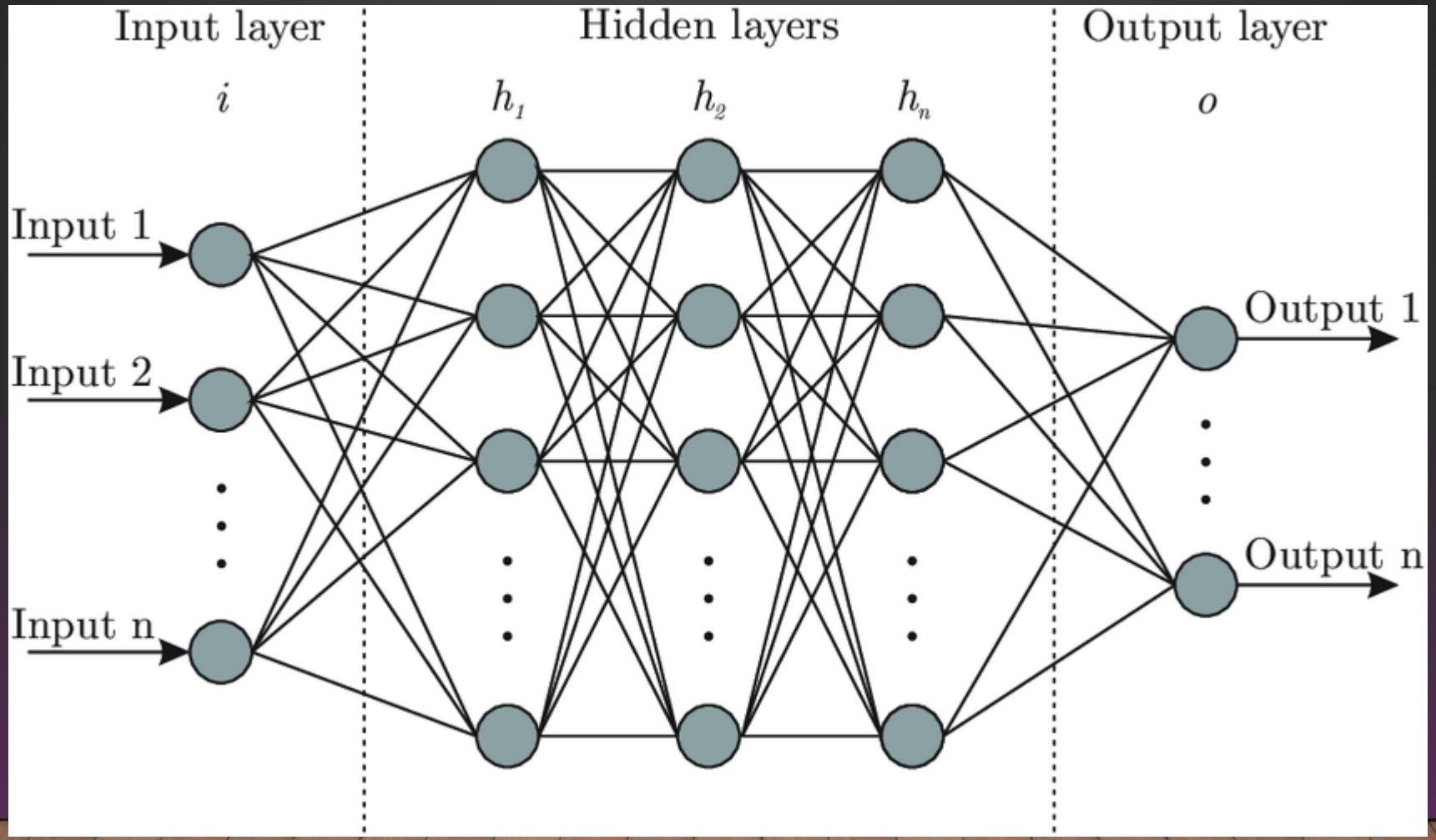
Additional methods used for CNN:

- Input is of shape 28x28x1
- Feature Detector of size 3x3
- Max pooling with 2x2 filter size
- 32, 64, and 128 size feature detector is used.

ANN MODEL DESCRIPTION

- Our model is sequential in nature since we are initializing this deep learning model as sequence of layers . The information is propagated from input layer to hidden layer to output layer through the model.
- Input layer of our network expects rows of data with 784 variables .We have input dimension as 784 which is equal to the number of columns in the dataset .
- We have used four hidden layers each of which has 404 nodes and uses the Rectified Linear Unit (ReLU) Activation Function because of its fast computation capabilities.
- ‘Dense’ at each layer signifies that layers will be fully connected in our model .
- Dropout Layer and BatchNormalization layers have been used in the model.
- At output layer , we have 25 nodes and we have used Softmax activation function.

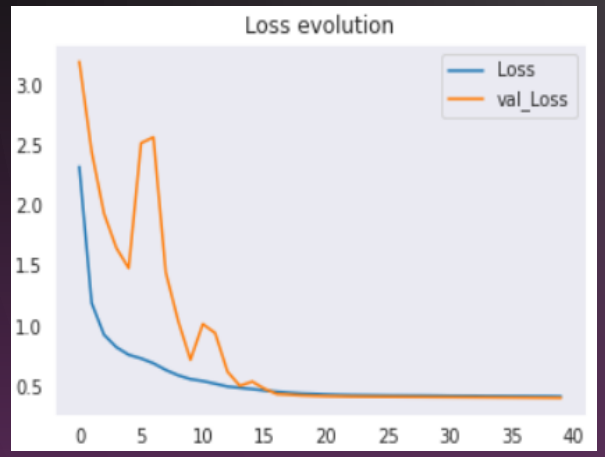
Architecture of ANN

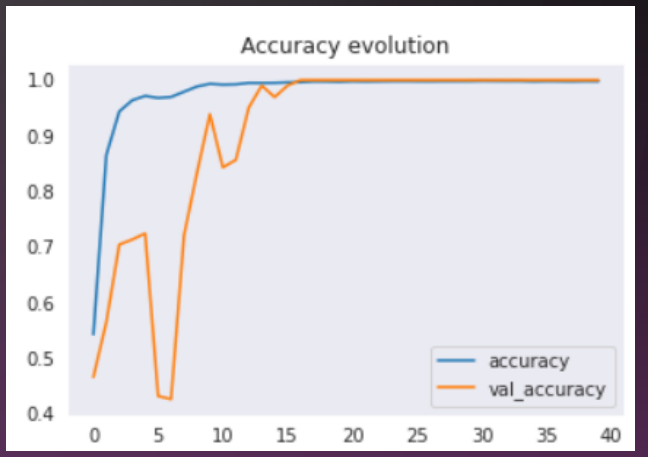


RESULTS AND ANALYSIS FOR ANN

- After training our model on 40 epochs with a batch size of 512.The maximum accuracy obtained on the training set was 99.9% and in validating set was 100%.
- The accuracy and AUC score obtained on the test set is low. Thus, the model is overfitting.

	Accuracy	Precision	Recall	AUC Score	F1 Score
Ours	0.8353	0.83	0.82	0.9142	0.82

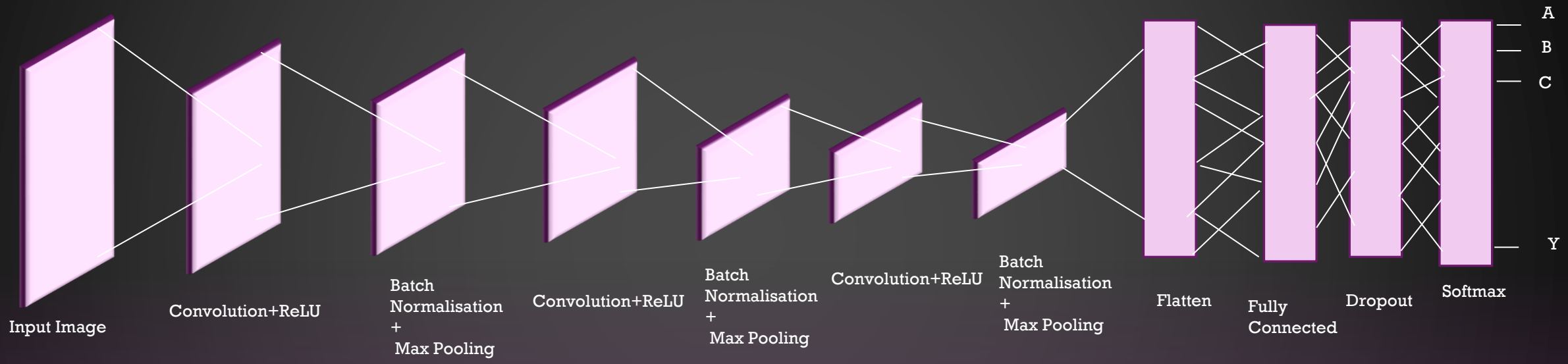




CNN MODEL DESCRIPTION

- Our model is sequential in nature since we are initializing this deep learning model as sequence of layers . The information is propagated from input layer to hidden layer to output layer through the model. This network employs a mathematical operation called convolution.
- After convolution operation ReLU activation function is applied to increase non linearity in our CNN model.
- Now BatchNormalization is used to keep the output of the layer of the same range to get an unbiased result.
- After this max pooling operation is performed. Max pooling helps to get rid of unnecessary features and moreover account for their spatial or textual or any kind of distortion.
- After first convolution network, the output was send to more convolution network in which number of feature detectors were 64 and 128 respectively.
- After this flattening operation is performed to make it suitable for fully connected layer's input.
- The Dropout layer is applied with a dropping rate of 25% which is fed to the dense layer with 25 neurons for 25 different classes.
- Now the output of this hidden or dense layer layer is passed through the softmax activation function which converts input linear data into probability array of 25 classes.

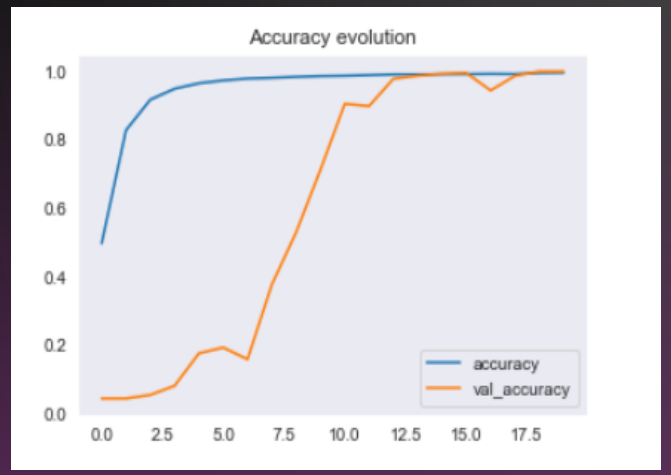
ARCHITECTURE OF CNN

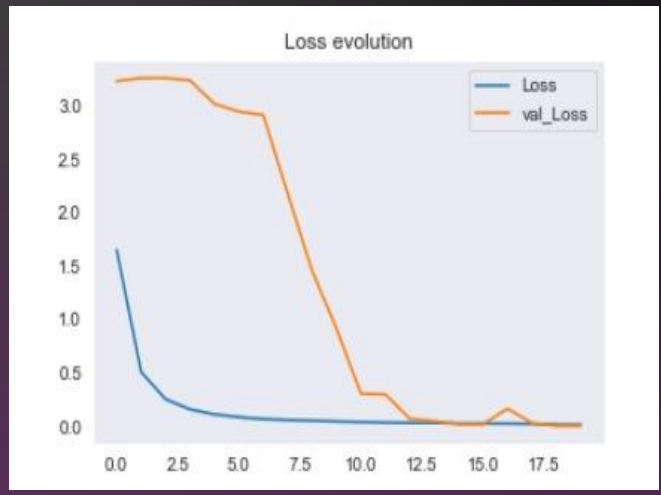


RESULTS AND ANALYSIS FOR CNN

- After training our model on 20 epochs with a batch size of 512.The maximum accuracy obtained on the training set was 99.60% and in validating set was 99.95%.
- Some letters had difficulty in prediction, particularly due to the similarity in the appearances of the letters. For example, O and S are confused with each other since the fingers are positioned in the same way with only a slight difference in the way the fingers are clenched, and the position of the thumb.

	Accuracy	Precision	Recall	AUC Score	F1 Score
Ours	0.9963	0.99	1.00	0.9981	1.00





CONCLUSION

- ANN model achieved an accuracy of 83.53%. The accuracy obtained is low as the model overfitted.
- We used various techniques to remove overfitting in our model which include Batch Normalization, dropout layers, Regularization, and early stopping.
- Despite of applying all these techniques, the ANN model did undergo overfitting and that's why we decided to opt CNN to train our model.
- Our proposed CNN Model was able to perform with 99.63% accuracy on our test dataset.
- When using only one convolutional layer there was overfitting in the dataset and to overcome this issue of overfitting, adding two more layers so that we dropped some parameters and still preserve the essential features of images.
- Moreover, dropout layer, BatchNormalization and data augmentation was used to overcome the same and achieve a better performance of our network.
- Future study may extend our work to accept video frames to include letters j and z in the classification so that more varied inputs can be processed and understood by the network.
- Furthermore, there is a need for a large public dataset for automatic sign language recognition to utilize new deep learning techniques and a better way to benchmark performance.

REFERENCES

[1] Srinath S, Ganesh Krishna Sharma, “Classification approach for Sign Language Recognition”, International Conference on Signal, Image Processing, Communication & Automation, 2017.

[2] Shivashankara S, Srinath S, “A comparative Study of Various Techniques and Outcomes of Recognizing American Sign Language: A Review”, International Journal of Scientific Research Engineering & Technology (IJSRET), Vol. 6, Issue 9, pp.1013-1023, 2017.

[3] Shivashankara S, Srinath S, “A Review on Vision Based American Sign Language Recognition, its Techniques, and Outcomes”, 7th IEEE International Conference on Communication Systems and Network Technologies (CSNT-2017), pp.293-299, 2017.

[4] S. Naidoo, C. Omlin, and M. Glaser, “Vision- based static hand gesture recognition using support vector machines,” University of Western Cape, Bellville, 1998.

[5] C. Vogler and D. Metaxas, “Parallel hidden markov models for american sign language recognition,” in Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, vol. 1. IEEE, 1999, pp. 116–122.

[6] Pei Xu, A real time hand gesture recognition and human-computer interaction system, In: Proceeding of the Computer Vision and Pattern Recognition, 2017

[7] J. Singha and K. Das, ”Indian Sign Language Recognition Using Eigen Value Weighted Euclidean Distance Based Classification Technique”, International Journal of Advanced Computer Science and Applications, vol. 4, no. 2, 2013..

THANK YOU