**Interpreting Doctor’s Handwritten Prescription Using Deep Learning Techniques**

Rizwanullah Mohammad, Ajay Kumar Varma Nagaraju and Suneetha Manne

Department of Information Technology, Velagapudi Ramakrishna Siddhartha Engineering

College, Andhra Pradesh, India

[mailtorizwanullah@gmail.com](mailto:mailtorizwanullah@gmail.com)

[ajaynagaraju32@gmail.com](mailto:ajaynagaraju32@gmail.com)

[hodit@vrsiddhartha.ac.in](mailto:hodit@vrsiddhartha.ac.in)

**Abstract:** A Doctor’s Handwriting Recognition model can predict (recognize) the text present in the doctor’s prescription, by feeding image of that medicine name as an input to the model and the model processes the image with deep neural network and it predicts the text present in the image and it gives the final medicine name as digital text. This model is suitable only for Text written in English Language and not suitable for other languages of texts written in prescription. The model based on training dataset the output it produce may get varied and based on training images count. Both convolution layers and Bi-Directional LSTM layers can be used for feature extraction and recognizing text respectively. Apart of that, we made a comparison between Bi-Directional LSTM’s and Bi-Directional GRU’s Performance.

**Keywords:** Bi-Directional LSTM Layers, Bi-Directional GRU, Convolution Layers, Adam optimizer, Relu Activation Function.

**1 Introduction**

It is most common that people can’t understand and interpret the doctor’s handwriting. The calligraphy they follow which is always challenging for ordinary people and even for pharmacist to understand doctor’s handwriting. Until they understand correctly then cant give correct medicine to Patient. Due to usage of wrong medicines they may face severe consequencies with respect to their health. This problem need to be solved with the latest technologies we are having at present. The solution for this is deep learning models. A deep learning model can take large input of data and can process with help of neural network and layers. They can give high accuracy and more reliable. Now with the help of deep learning techniques involving all the terms in deep learning to provide a optimal solution for this. The Bi-Directional LSTM model can provide a solution which can predict text present doctor’s prescription’s image which we passed as input to our model.

**2 Literature Study**

Here are the available models, That Researchers proposed and developed. They Proposed different methodologies to recognize text in an image and showcasing their performance.

Handwriting recognition for medical prescriptions using a cnn-bi-lstm model (2021) [1]:

Authors presented a model employing the BI-LSTM Model for the recognition of a doctor's handwriting. They have only created a model; no mobile or web applications have been created to execute the model in real time. In order to minimise overfitting and increase the model's resistance to noise, they employed data augmentation approaches. The feature extraction is carried out using Convolutional Neural Networks with many layers, and the decoding of the extracted features into English letters is assisted by Bi-LSTMs.

Recognition of doctors’ cursive handwritten medical words by using Bidirectional lstm and srp data augmentation (2021) [2]:

Authors presented a model for the identification of doctors' handwriting. After employing the SRP Augmentation approach, they achieved an accuracy of 89%. Some of the participating physicians' prescription pictures were made available. A collection called the "Handwritten Medical Term Corpus" was produced with 17,431 handwritten examples of 480 medical terms in English and Bangla. The introduction of SRP, a data augmentation technique, increased the size of the data sets. For predicting the handwriting of doctors, an online character recognition system utilising Bi-LSTM was employed.

Doctor’s cursive handwriting recognition system using deep learning (2019) [3]:

Authors presented a model for interpreting the doctor's handwriting was developed. They employed CRNN Model. They created a smartphone application that allows users to enter images and receive digital text as output. Of the 540 input photos, 389 have been successfully recognised.The accuracy of the tests conducted using the mobile application was 72%. The model is implemented through the use of a mobile application called DCHRS, which stands for "Doctors' Cursive Handwriting Recognition System" and aims to recognise the name of the medication inside the image of doctors' cursive handwriting that has been captured, as well as to provide the normal text version of the handwriting.

Handwriting recognition using Deep Learning in Keras (2018) [4]:

Authors utilised the Mnist dataset, which contains numbers, to construct a model to identify handwritten text in an image. Comparisons between CNN and Multi-Layer Perceptrons were conducted. They demonstrated that CNN provides better accurate results than Multi-Layer Perceptrons. The accuracy was 99% with CNN with 20 epochs, but only 90% with the same Multi-Layer Perceptrons. They came to the conclusion that CNN offers more accuracy.

**3 Proposed Work**

**3.1 Dataset**

The proposed model was trained with 86800 images in gray scale. We used IAM Dataset which contains handwritings of different people with different fonts. For training we splitted dataset into 90:5:5 Ratio. 90% of training data, 5% of validation data and 5% of testing data.

**3.2 Pre-processing**

In this pre-processing phase the images will get reshaped to width of 128 and height of 32 and padding to 99. And then the datatype will be changed to float32 which we call it as casting. Thıs may increase the model performance.This is distortion free image processing. Initial, using tf.image is the first stage in the resizing process. We should send the picture path, width, and height to the resize() function while maintaining the aspect ratio. Following this, we will add additional padding to resized image. Padding can be added to image by subtracting width and height values(128,32 respectively) with image shape which we want to add padding to, with help of tf.shape() function. With tf.transpose() function by giving perm = [1,0,2] we are setting up the required tensor dimensions. Perform flip\_left\_right of image inorder to get image flipped along with width dimensions.

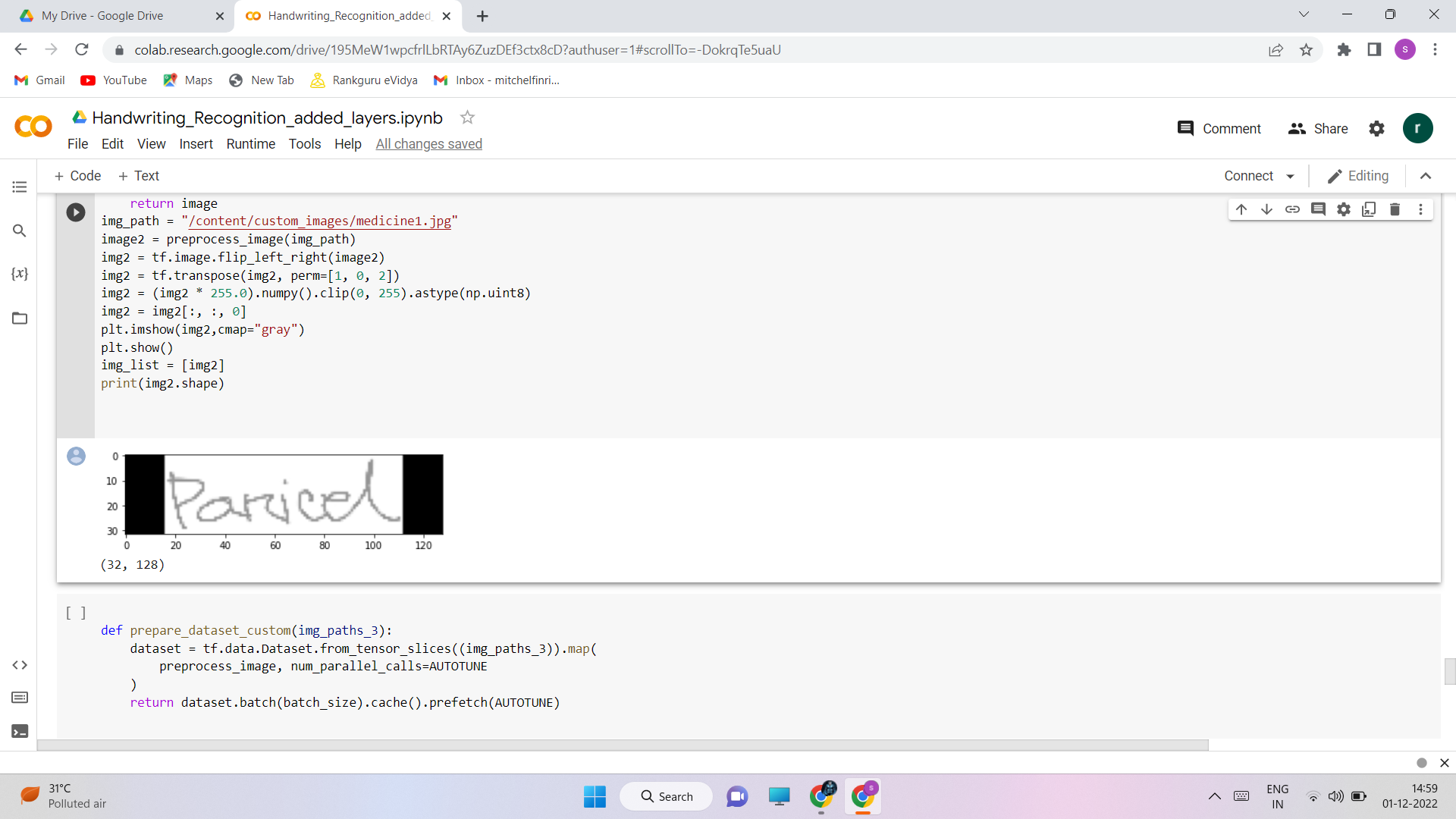


Fig. 1. Distortion free image pre-processing

**3.3 Design Methodology**

The below diagram describes architecture of our work.

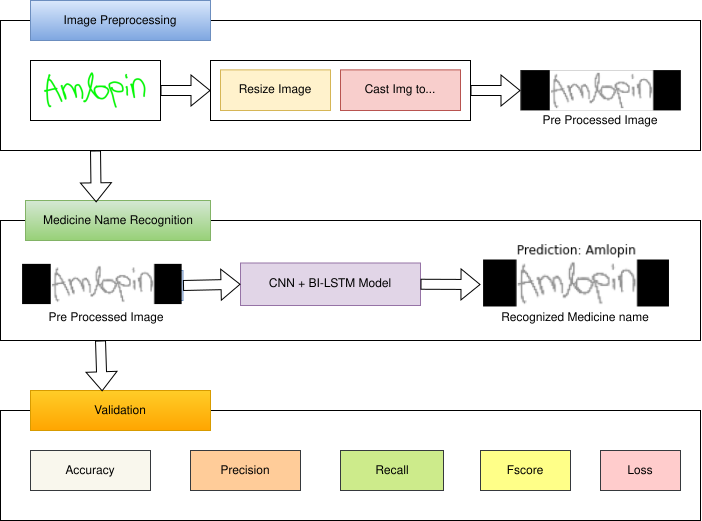


Fig. 2.Architecture Diagram for our work

We utilised cropped images from the IAM Dataset and a few medical datasets in conjunction with our model to train. The training dataset, validation dataset, and testing datasets have been divided in a ratio of 90:5:5.The network has a sophisticated design that includes seven convolutional layers, optional batch Normalization layers, Max Pooling layers, Relu activation functions, a Bi-directional LSTM layer, and a CTC layer.The procedure increases the number of channels in the first convolution layer from 1 to 64. which, after several layers, is raised to 128. The Bi-directional LSTM layer aids in the decoding of the convolution layer-generated feature mapping. We will next create a layer for label input for the appropriate images, followed by a thick layer. To find the CTC loss, the last layer would be the CTC layer. The next step is to determine the callback distance. During this stage, ctc decoding will be done using predictions.Then,using tf.sparse.from\_dense and dtype as an int64, sparse the predictions made from dense. Identify a point when an increase in epoch values leads to a rise in loss value and the model will no longer improve performance at which point training may be stopped. Next, build the model and train it with various rising epoch values, setting checkpoints as necessary.We can use the count variable to determine the model's accuracy by passing some data as input, checking the total number of correct predictions, and calculating the percentage of right predictions.The accuracy will be proportional to the initial weights we are establishing in the input layer. Setting appropriate starting weight values for the input layer and subsequent layers, as well as choosing an appropriate activation function, are required. Relu inside a convolution layer and Dense layer are typically utilised as activation functions for models like CNN and LSTMs.

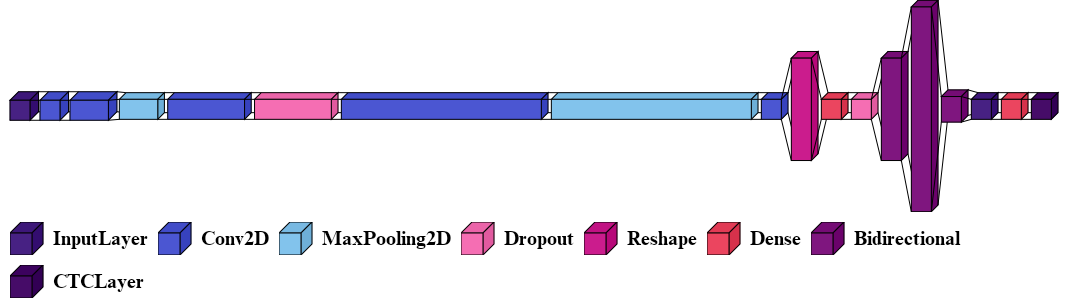


Fig. 3. Our Model design for Bi-Directional LSTM

Fig(3) shows model design for Bi-Directional LSTM Model. The Bidirectional layer from fig(3) represents the Bi-Directional LSTM Layer. The initial layer is input layer of height 32, width 128 and the channel is 1 because its a grayscale image. Next upcoming 2 layers are convolutional layers of filter size 32,128 respectively. Next, from the above Figure we can identify different layers with filter sizes given as 3rd parameter for CNN layers. For Bi-Directional LSTM layers number of hidden cells are 512,1024,64. The CTC\_LOSS layer computes the character wise error rate instead of word wise error rate and returns loss value for every step per epoch.

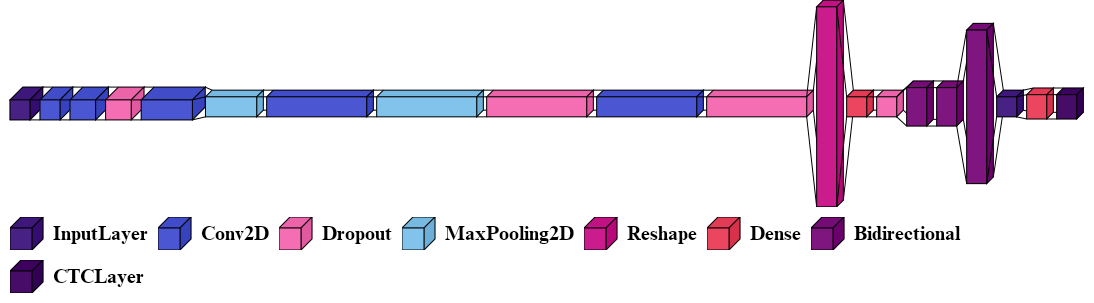


Fig. 4. Our Model design for Bi-Directional GRU

Fig.4 shows model building phase with Bi-Directional GRU. The Bidirectional in Fig(4) represents Bi-Directional GRU Layer. The input layer parameters are same as how its done for Bi-LSTM model and we can notice that there are some dropout layers in between convolutional and pooling layers in order to control overfitting of model. There are total of 5 convolution layers of filter sizes 32,128,256,512,1024 respectively. And next there are 3 Bi-Directional GRU layers with hidden cells count 128,128,512 respectively. CTC\_LOSS working is same as we seen in Bi-Directional LSTM model.

When we compare both models, Bi-Directional LSTM is having more number of parameters for training than Bi-GRU. It depends on our model’s number of hidden units. Considering individual characters error rate is necessary in order to get good efficient and accurate model.

**3.4 Selection of Algorithm**

We implemented our model with two different algorithms, they are Bi-Directional LSTM and Bi-Directional GRU. The CNN is common in both models. But the type of RNN used is different. We used CNN + Bi-Directional LSTM in first model and CNN + Bi-Directional GRU in second model. Here, we are trying to make comparison between this two RNN’s which is performing well. From Fig. 2 in medicine name recognition phase the algorithm we used are different,but other steps remain same.Both LSTM and GRU are Recurrent neural networks but the difference is the size of data they can handle. We are using IAM dataset and training data size is 86,800 Images which is large size.So, inorder to handle this large data among all RNN’s the Bi-Directional LSTM is best and the next one is GRU. If the dataset size is small then GRU is most preferred.The gates present in LSTM are Input,Output and Forget. Whereas in GRU the gates present are only update and reset. We can tell that GRU is less complexive as compared with LSTM. So, its better to use GRU for small sized data and LSTM for large sized data. We have selected both algorithms to test which algorithm will perform well.

The Mathematical Notation of Bi-Directional LSTM as follows:

, ] + (1)

From equation 1, The represents the output and represents the output at time t. The represents the hidden layer function. denotes the hidden layer weights matrix. And similarly denotes hidden layer vector for bias values.gives the forward hidden sequence at time t and gives the backward hidden sequence at time t. As we are employing Bi-directional layers, we can have two directional sequences. Here, the iterations are done on forward and backward sequences and then the output layer will get updated.

The Mathematical Notation of Bi-Directional GRU as follows:

(2)

Similar to Bi-Directional LSTM,the Bi-Directional GRU also allows data sequence in two directions forward and backward.From Equation 2, is a GRU function that denotes the data sequence flow in Forward direction and is a GRU function that denotes the data sequence flow in Backward direction. is vector concatenation operator for and Data sequence flows.denotes forward GRU’s state and denotes backward GRU’s state. denotes input vector.denotes output of cell at time t. By performing concatenation between forward and backward GRU states gives the output .

**4 Result and Observations**

**4.1 Test case Results**

Below are our model’s testcase results.We passed some custom images to model and we got the below predictions.

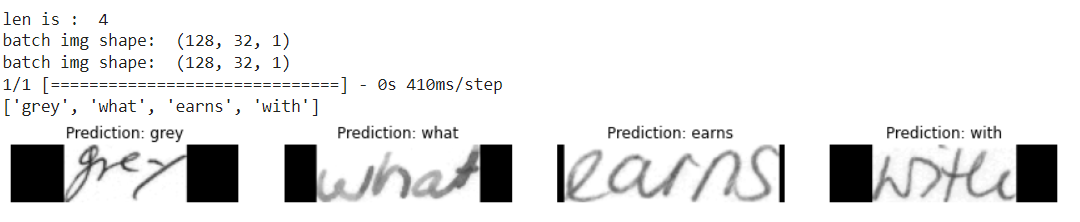


Fig. 5. Output Test Case predicted by the model

The model was trained for 25 Epochs and we got this predictions. We passed 4 images custom image inputs from testing data in IAM Dataset and we got 4 images predicted correctly.Firstly,we have done preprocessing image as given in sub chapter 3.2 and passed as input to model to predict them.

**4.2 Observations and Analysis**

We noted our observations of Accuracies with two different models i.e Bi-Directional LSTM and Bi-Directional GRU at different epochs and displayed in below table.

Table. 1. Comparison of Bi-LSTM and Bi-GRU Accuracies

|  |  |  |
| --- | --- | --- |
| Epochs | Accuracy (%) | |
| BI-LSTM | BI-GRU |
| 20 | 75 | 69 |
| 25 | 78 | 74 |
| 30 | 81 | 77 |

From the above table, we trained two models with Bi-Directional LSTM and Bi-Directional GRU Algorithms seperately. The accuracies obtained are noted in above tabular form. At 30th Epoch Bi-Directional LSTM gave 81% Accuracy and where as at same epoch the Bi-Directional GRU gave only 77% Accuracy. It took around 5 hours to train the model upto 30 epochs with Bi-Directional LSTM Algorithm and for training Bi-Directional GRU upto 30 epochs it took 3 hours of time. Bi-Directional GRU which is less complexive than Bi-Directional GRU model got completed 100% training of 30 epochs within lesser time as compared to Bi-Directional LSTM, but Bi-GRU gave lesser Accuracy than Bi-LSTM. Where there is larger sized datasets the Bi-Directional LSTMs are performing well than Bi-Directional GRUs.

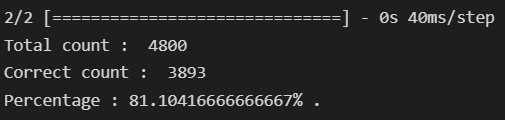


Fig. 6. Accuracy of Bi-Directional LSTM Model.

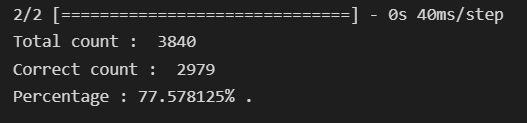


Fig. 7. Accuracy of Bi-Directional GRU Model.

From Fig(6) and Fig(7) we can observe the accuracies obtained for Bi-Directional LSTM and Bi-Directional GRU models respectively. We took 88% training data for Bi-Directional GRU model.and 90% of training data for Bi-Directional LSTM model. Controlling validation loss is important to avoid overfitting of model. By adding sufficient dropout value overfitting can be controlled and also size of data we are using for validation may vary the Validation loss accordingly. We can notice from above 4,5 figures, the accuracies of two models respectively.

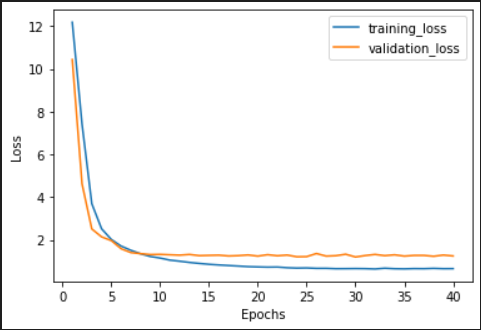


Fig. 8. Loss vs Epochs with Bi-Directional LSTM

The Figure 8 shows the graph plotted between Epochs and Loss values. These Loss values are obtained when we trained our model with 40 epochs and saved the best weights with the help of checkpoints. At 30th epoch the training and validation loss are minimum. The training loss is 0.6 and validation loss is 1.24.

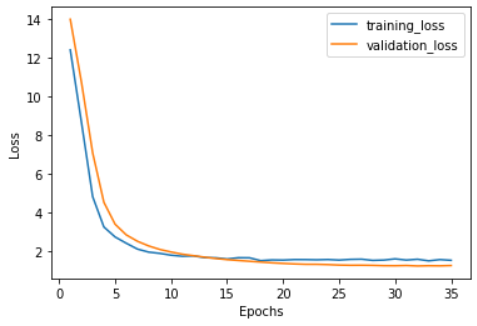


Fig. 9. Loss vs Epochs with Bi-Directional GRU

The Fig 9 shows loss vs epochs graph with Bi-Directional GRU model. We trained model upto 35 epochs and we saved best weights that is at 30th epoch we observed that model is giving minimum training and validation loss. 1.5 and 1.3 are training and validation losses we got with Bi-Directional GRU.With Bi-Directional LSTM model we got 81% Accuracy,0.96 Precision value,0.83 Recall value and the harmonic mean of precision and recall gives us F-score that is 0.89.

And with Bi-Directional GRU we got 77% Accuracy,0.95 Precision value,0.80 Recall value and F-score of 0.86.By seeing this Metrics we can clearly notice that Bi-Directional LSTM is performing well with large datasets for training and validation.As LSTMs are having more gates than GRUs, always the algorithm which we want to use, if its having more gates then its more complexive but it can perform better than others which have less gates.

**5 Conclusion and Future Work**

This model helps Pharmacists and normal people to recognize the medicine name accurately which is present in the Doctor’s Handwritten Prescription. This effectively offers text in all handwriting typefaces. The feature extraction is carried out by Convolutional Neural Networks with many layers, and the decoding of the extracted features into English letters is assisted by Bi-LSTMs.We employ CTC to circumvent the fact that the true alignment between the input and the output is unknown. To accurately identify language specific to prescriptions provided by the doctors, more bias is applied to words that are present in a manually produced corpus. The accuracy can be maximised by increasing the size of the training data. And also when we compare Bi-Directional GRU and Bi-Directional LSTM, the Bi-Directional LSTMs are performing well and more accurate when dataset is large.

**Future work**: The Accuracy for this model can be further improved by training with more handwritten prescriptions. Further an API can be created for this model which can be useful Mobile Applications or web application to use this model for Recognizing text from Prescription or from input of cropped Medicine Name. And more layers can be added by keeping complexity of model in mind. Training same dataset with new and Advanced Deep Learning Algorithms or Advanced Neural Networks can improve Accuracy.

**References**

1. T. Jain, R. Sharma and R. Malhotra, "Handwriting Recognition for Medical Prescriptions using a CNN-Bi-LSTM Model," 2021 6th International Conference for Convergence in Technology (I2CT), 2021, pp. 1-4, doi: 10.1109/I2CT51068.2021.9418153. (T. Jain, 2021)
2. S. Tabassum et al., "Recognition of Doctors’ Cursive Handwritten Medical Words by using Bidirectional LSTM and SRP Data Augmentation," 2021 IEEE Technology & Engineering Management Conference - Europe (TEMSCON-EUR), 2021, pp. 1-6.
3. L. J. Fajardo et al., "Doctor’s Cursive Handwriting Recognition System Using Deep Learning," 2019 IEEE 11th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management ( HNICEM ), 2019, pp. 1-6.
4. España-Boquera, S., Castro-Bleda, M.J.”A Spanish dataset for reproducible benchmarked offline handwriting recognition”. Lang Resources & Evaluation 56, 1009–1022 (2022).
5. Maalej, R., Kherallah, M. New MDLSTM-based designs with data augmentation for offline Arabic handwriting recognition. Multimed Tools Appl 81, 10243–10260 (2022).
6. A. Harikrishnan, S. Sethi and R. Pandey, "Handwritten Digit Recognition with Feed-Forward Multi-Layer Perceptron and Convolutional Neural Network Architectures," 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), 2020.
7. A. Nikitha, J. Geetha and D. S. JayaLakshmi, "Handwritten Text Recognition using Deep Learning," 2020 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT) (2020).
8. M. I. Swindall et al., "Exploring Learning Approaches for Ancient Greek Character Recognition with Citizen Science Data," 2021 IEEE 17th International Conference on eScience (eScience), 2021.
9. S. Hassan, A. Irfan, A. Mirza and I. Siddiqi, "Cursive Handwritten Text Recognition using Bi-Directional LSTMs: A Case Study on Urdu Handwriting," 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML), 2019.
10. ul Sehr Zia, N., Naeem, M.F., Raza, S.M.K. et al.”A convolutional recursive deep architecture for unconstrained Urdu handwriting recognition”. Neural Comput & Applic 34, 1635–1648 (2022).
11. Alrobah, N., Albahli, S.”Arabic Handwritten Recognition Using Deep Learning: A Survey”. Arab J Sci Eng 47, 9943–9963 (2022).
12. U. Shaw, Tania, R. Mamgai and I. Malhotra, "Medical Handwritten Prescription Recognition and Information Retrieval using Neural Network," 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC), 2021, pp. 46-50
13. S. Sharma and S. Gupta, "Recognition of Various Scripts Using Machine Learning and Deep Learning techniques-A Review," 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC 2021).
14. Sethy, A., Patra, P.K., Nayak, S.R. “A Deep Convolutional Neural Network-Based Approach for Handwritten Recognition System”.Advances in Intelligent Systems and Computing, vol 1349 (2022).
15. Altwaijry, N., Al-Turaiki, I. Arabic handwriting recognition system using convolutional neural network. Neural Comput & Applic 33, 2249–2261 (2021).