2021 6th International Conference for Convergence in Technology (I2CT)
Pune, India. Apr 02-04, 2021

Handwriting Recognition for Medical Prescriptions using a CNN-Bi-LSTM Model

Tavish Jain
Department of Computer Science & Engineering
Delhi Technological University
New Delhi, India
jaintavish@gmail.com

Rohan Sharma
Department of Computer Science & Engineering
Delhi Technological University
New Delhi, India
rohan_dce@outlook.com

Ruchika Malhotra
Department of Computer Science & Engineering
Delhi Technological University
New Delhi, India
ruchikamalhotra@dtu.ac.in

2021 6th International Conference for Convergence in Technology (I2CT) | 978-1-7281-8876-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/I2CT51068.2021.9418153

***Abstract* - It is commonly seen that it is tough to read the handwritten text from medical prescriptions. It is mostly due to the different style of handwriting and the use of Latin abbreviations for medical terms which is usually unknown to the general public. This can make it difficult for both patients and even pharmacists to read the prescription, which can have negative or even fatal consequences if read incorrectly. This paper demonstrates the use of a CNN-Bi-LSTM model along with Connectionist Temporal Classification. The prescribed model consists of three components, the convolutional layers for feature extraction, the Bi-LSTM network for making predictions for each frame of the context vector and the final decoding to translate each character in the recognized sequence by LSTM layers into an alphabetic character using the CTC loss function. A linear layer is added after the bi-LSTM layer to compute the final probabilities, which will be decoded. We also built a corpus manually containing the terms widely used in the medical domain, commonly used in prescriptions. We then use string matching algorithms, and string distance functions to find the nearest word in the corpus, so that bias is given to medical terms for increasing accuracy of the predicted output.**

Keywords - Long-short term memory networks, convolutional networks, neural networks, connectionist temporal classification, recurrent neural networks, character error rate, batch normalization, Seq2Seq networks, Adam Optimizer, PyTorch

I. INTRODUCTION

It is becoming increasingly common that people incorrectly read the medical prescriptions, and hence go on to consume wrong medicines or wrong dosage of medicines which is very harmful to their health, and may prove to be fatal in some cases. This mostly happens due to the fact that most doctors have illegible handwriting, and also due to the lack of medical knowledge of patients and the chemists. This is becoming an increasingly common problem, but can be solved using technology.

Deep Learning has been a major force in driving research advancements around text recognition [3]. Deep learning models have been a success due to the recent architectures and availability of large scale annotated data. There have been many attempts to leverage the power of deep learning to solve this issue in the past. But with the recent advancements in the field of Deep Learning, text recognition has become highly accurate and reliable which is a great solution to this problem. There are various different existing methods for simple text recognition, but the need of the hour is a custom technique specially suited for reading medical prescriptions. Such a technique could help remove errors in reading the medicine or treatment names and dosage, and thus help save people's lives.

Through this paper we intend to develop a technique that is specially trained to recognize medical prescriptions correctly. The technique will take images of medical prescriptions as input, and return the text written in the prescription so that less mistakes are made in reading the prescription. The general steps that were used in the paper for handwriting recognition were sequenced as:

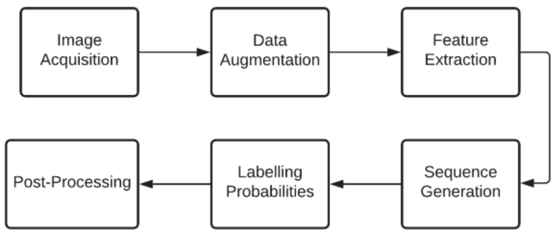

Fig. 1. Flow of logic

We have also included the methods used for building the dataset, writing the python code, the training process for the model and the results that were finally achieved.

II. LITERATURE REVIEW

A lot of researchers have researched into the topic of text recognition, and various approaches follow, which include Keyword Spotting [1], text orientation detection, information energy based on each pixel, stroke identification, MSER techniques, and also Deep Learning techniques that involve convolutional neural Networks [6].

Abhishek Bal et al. [8] presented a handwritten document analysis which uses segmentation and detecting the amount of pressure applied for the documents. The method is based on horizontal and vertical projections that divide the line and words. The technique also performs well in the presence of skewed and overlapped text. The method was tested on the IAM database.

Kanchan Keisham et al. [9] proposed a line segmentation approach on information energy that is calculated for all pixels, and the classification is done with the help of Artificial Neural Networks.

Nibaran Das et al. [4] demonstrated the use of the convex hull algorithm. A total of one hundred twenty-five features are extracted by the use of various attributes of the hull.

These experiments were carried out on the Bangla basic characters' dataset.

Nafiz Arica et al. [5] put forward recognition algorithms, which aimed to recognize cursive handwriting. The segmentation procedure included converting the image to grayscale, and then applying Hidden Markov Models for prediction of the characters.

Subhadip Basu et al. [7] presented the use of multi-layer perceptrons for recognition. Feature sets were designed for character recognition and used three kinds of topological features. These experiments were carried out on the Bangla basic characters' dataset.

Namrata Dave et al. [10] proposed techniques that could help segment the text. Three different levels of segmentation were proposed to be used. First a text level segmentation is done, followed by a word-level segmentation, and finally a character level segmentation has been explained.

III. PROPOSED ARCHITECTURE

A. Data Preparation

We used the publicly available IAM dataset for this paper. We registered on the website and downloaded images of lines, and its annotations, which were available in XML format, as well as the TXT format. In the images, there were "bounding boxes" around the words which theoretically gave additional context for a neural network to learn. We augmented the input images by distorting it. We pass in the complete images to the neural network, and its annotations in an encoded format, by creating a dictionary of all the characters that were used in the recognized text. We pass in the image of a line of textual data, along with the image, which is later on decoded when the model returns its output.

B. Model Architecture

We try to convert the input image into the text using a deep convolutional neural network, which converts the input image into a context vector, which is then sent as the input to the Bi-LSTM Decoder network, which outputs the predicted and converted sequence from the image. The network uses a complex architecture, using seven convolutional layers, along with optional batch Normalization layers, Max Pooling layers, ReLU and LeakyReLU activation functions for the Encoder, and a Bi-directional LSTM layer and a Linear Layer as the Decoder, which finally returns us the predicted probabilities.

The built network takes in a variable width image as an input, where the length of the image is sixty pixels. The variable dimension of the images is normalized after the first convolutional layer. The first convolution operation changes the number of channels in the image from three to sixty-four which is increased up to five hundred twelve in further convolutional layers. Every layer is followed by ReLU, MaxPool layers and optional Batch normalization layers. This marks the end of the Encoder, which takes in the input as a processed image, and returns the context vector. Context Vectors can be said to be fixed-length vector representations which store model weights from the Encoder layers, and are fed as an input to the Decoder layer. The Decoder consists of two Bi-directional LSTM layers, along with a dropout value of 0.5. The Decoder is finally ended by adding a Linear Layer at the end, which has its count of input nodes as two thousand forty-eight, and the count of output nodes as the number of characters in the dictionary. The final Linear layer acts as an embedding layer. The final Linear layer gives us the output probabilities, which are of the shape [BATCH_SIZE, DICTIONARY_SIZE, SEQUENCE_LENGTH], which is then passed to decode to characters.

To decode the output probabilities, we use the argmax function to find the index of the maximum probabilistic index, and the index is returned as a vector, which is then converted to the corresponding English character. Sequences of the returned English characters are accumulated together to eventually form the predicted text from the medical prescription image. from the mapping initially created. This however contains a lot of extra characters, which is then normalized by the use of Connectionist Temporal Classification. The neural network model can be trained end to end using widely available IAM Handwriting dataset. We also use data augmentation on input images by distorting the image, adding meshes to the image, applying linear and cubic interpolation methods and finally warping the image. Since the Encoder is fully convolutional, it is not restricted to fixed-size input.

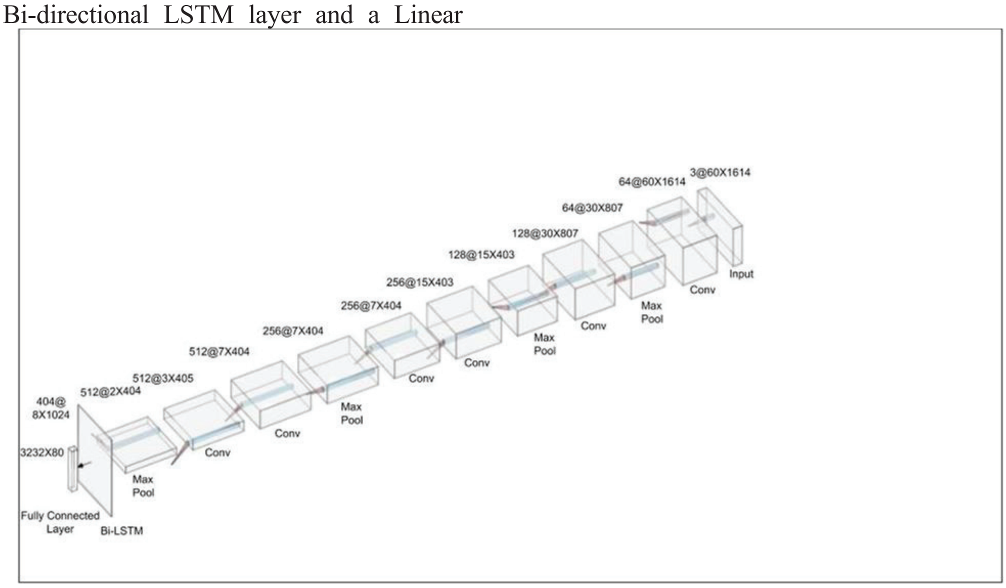

Fig. 2. Model Architecture

C. Predictions with Bi-LSTM

Long Short Term Memory Networks are used to recognize/classify the next character [2] from the input text.

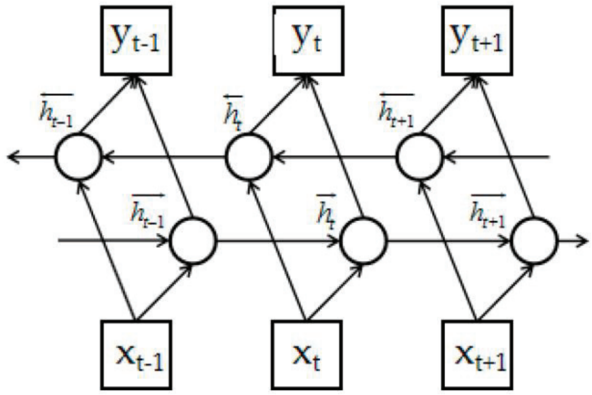

Fig. 3. Bi-LSTM Network Structure

The Encoder Layer returns us a tensor containing probabilities of occurrence of each character for a fixed sequence length. The argmax function is then used to find the character having the maximum probability, which is then finally used in the output character sequence. This output character sequence is then decoded into alphabets by reverse mapping the indices selected by the argmax function, to the corresponding alphabets. The Connectionist Temporal Classification technique is then used to remove all the

D. Corpus for Medical Terms

We also manually built a corpus containing the medical terms, which are used in prescriptions. We use string matching algorithms and string distance functions to find the nearest word in the corpus, so that bias is given to medical terms for increasing accuracy of the predicted output.

IV. EXPERIMENTAL RESULTS

Training the deep learning model using more and more images will help in increasing the accuracy/ reducing the loss of the prescribed deep learning model. We used the *Large Writer Independent Text Line Recognition Task* which defines an experiment with well-defined training, test, and validation sets. This returns us with nine thousand plus training set images, which covers text-lines from over three hundred seventy writers, and thousand plus images for testing the built model, which covers text-lines from over one hundred twenty writers, all being mutually exclusive to each other.

We trained the model for thirty-two epochs. Training the model with more data and for longer epochs will help in increasing the accuracy of the model. We save the model configuration as and when we reach a minimum CER Loss value, and save the weights of the model. If the training loss comes out to be more than validation loss, it indicates that the model is under fitting. However, if the training loss is less than the validation loss, it indicates the model is overfitting. However, the pursued result is to have the training loss equivalent to validation loss. Training the model up to thirty-two epochs took six hours to finish.

The loss values as the model trained have been illustrated in the figure below. The orange curve denotes the test loss Vs Epochs. The blue curve denotes the training loss Vs Epochs of the model. Increasing the dataset collection will increase the accuracy rate.

Character Error Rate (CER) calculates the count of characters in the handwriting that the Deep Learning model did not read correctly. We prefer using CER as a metric, rather than WER (Word Error Rate), as it is not meaningful, as it would highly decrease the quality of the model. Predicting Words matching exactly without an error will be difficult as words could be of variable length with repeating characters, which might or might not be covered perfectly.

Logs are generated as epochs are completed showing training CER and test CER. The best model is saved/updated after every epoch. Some logs are attached below.

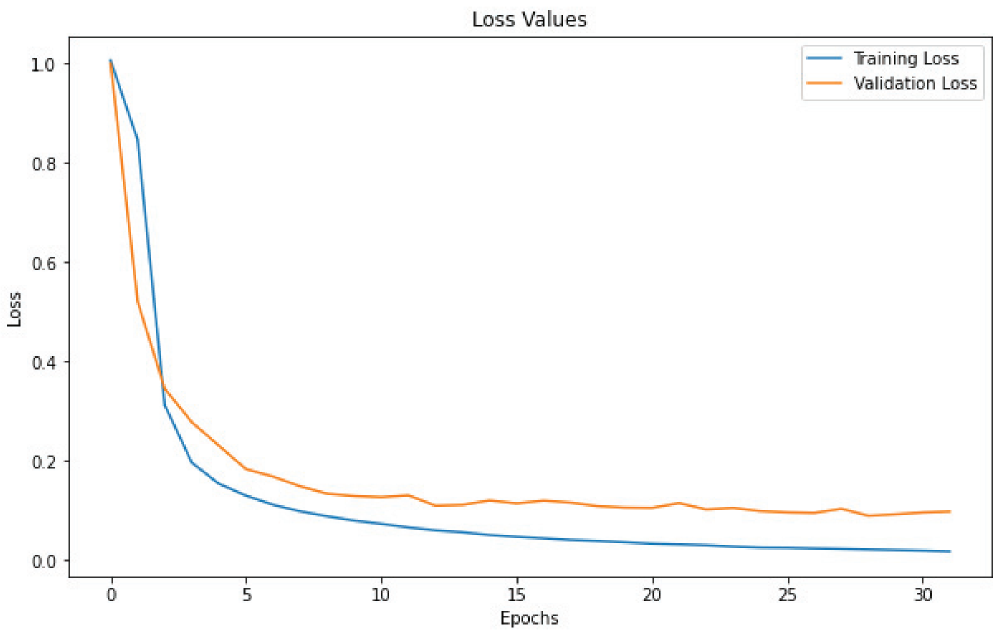

Fig. 4. Loss(CER) Vs Epochs

```
Starting for epoch:  18
Training CER 0.040290962858451795
Test CER 0.11566633122235549

Starting for epoch:  19
Training CER 0.038230602058216295
Test CER 0.1085081087030012
Saving Best
```

```
Starting for epoch:  28
Training CER 0.022497624425115867
Test CER 0.10314193567544347

Starting for epoch:  29
Training CER 0.021179178003325614
Test CER 0.08891598131148906
Saving Best
```

V. CONCLUSION

The paper deals with studying different techniques for handwritten text recognition. We've used data augmentation techniques to make the model more robust to noise, and also avoid overfitting. Multiple layers of Convolutional Neural Networks perform the feature extraction, and bi-LSTM's help in decoding the extracted features to English characters. Since the actual alignment between the input and the output is not known, we use Connectionist Temporal Classification to get around not knowing that alignment. More bias is given to words that are present in a manually created corpus to accurately recognize text specific to prescriptions offered by the doctors.

VI. ACKNOWLEDGMENT

The authors are very much grateful to the Department of Computer Science & Engineering of Delhi Technological University for giving us the opportunity to work on Handwritten Recognition for Medical Prescriptions. Both the authors sincerely express their gratitude to Dr. Ruchika Malhotra for giving constant encouragement in doing research in the field of image processing

REFERENCES

[1] Partha Pratim Roy, Ayan Kumar Bhunia, Ayan Das, Prithviraj Dhar, Umapada Pal," Keyword spotting in doctor's handwriting on medical prescriptions", Expert Systems with Applications, Volume 76, 2017, Pages 113-128, ISSN 0957-4174

[2] P. S. Dhande and R. Kharat, "Character Recognition for Cursive English Handwriting to Recognize Medicine Name from Doctor's Prescription," 2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA), Pune, 2017, pp. 1-5, doi:10.1109/ICCUBEA.2017.8463842.

[3] N. Chumuang and M. Ketcham, "Model for Handwritten Recognition Based on Artificial Intelligence," 2018 International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP), Pattaya, Thailand, 2018, pp. 1-5, doi: 10.1109/iSAI-NLP.2018.8692958.

[4] Nibaran Das, Sandip Pramanik, Subhadip Basu, Punam Kumar Saha, "Recognition of handwritten Bangla basic characters and digits using convex hull based feature set", 2009 International conference on Artificial intelligence and pattern recognition(AIPR-09).

[5] Nafiz Arica, Student Member, IEEE, and Fatos T. Yarman-Vural, Senior Member, IEEE, "Optical Character Recognition for Cursive Handwriting", IEEE transactions on pattern analysis and machine intelligence, vol. 24, no. 6, june 2002.

[6] Kamalanaban, E. & Gopinath, M. & Premkumar, S. (2018). Medicine Box: Doctor's Prescription Recognition Using Deep Machine Learning. International Journal of Engineering and Technology(UAE). 7. 114-117. 10.14419/ijet. v7i3.34.18785.

[7] Subhadip Basu, Nibaran Das, Ram Sarkar, Mahantapas Kundu, Mita Nasipuri, Dipak Kumar Basu, "A hierarchical approach to recognition of handwritten Bangla characters", Elsevier -2009

[8] Abhishek Bala and Rajib Saha, "An Improved Method for Handwritten Document Analysis using Segmentation, Baseline Recognition and Writing Pressure Detection",6th International Conference On Advances in Computing Communications, ICACC 2016, 6-8 September 2016, Cochin, India, Elsevier-2016.

[9] Kanchan Keisham and Sunanda Dixit, "Recognition of Handwritten English Text Using Energy Minimisation", Information Systems Design and Intelligent Applications, Advances in Intelligent Systems and Computing, Bangalore, India, Springer-2016.

[10] Namrata Dave, "Segmentation Methods for Hand Written Character Recognition", International Journal of Signal Processing, Image Processing and Pattern Recognition Vol. 8, No. 4 (2015), pp. 155-164.

[11] U. Marti and H. Bunke. The IAM-database: An English Sentence Database for Off-line Handwriting Recognition. Int. Journal on Document Analysis and Recognition, Volume 5, pages 39 - 46, 2002.