Application of Fuzzy Matching Algorithms for Doctors Handwriting Recognition

Riya Patil
Electronics & Telecommunication Engineering
Vishwakarma Institute of Technology
Pune, India
riya.patil19@vit.edu

Prasad Peshave
Electronics & Telecommunication Engineering
Vishwakarma Institute of Technology
Pune, India
prasad.peshave19@vit.edu

Prof Milind Kamble
Electronics & Telecommunication Engineering
Vishwakarma Institute of Technology
Pune, India
milind.kamble@vit.edu

***Abstract*— Doctor's handwritten prescriptions are often known to be indecipherable. Uncertainty in medical terms can have dire consequences. A method to effectively recognize medicine names written in doctor's handwriting is proposed in this paper. A corpus of 600 images is compiled with the help of multiple doctors. An exhaustive list of 50 medicines is used for the same. Recognition is performed using the Convolutional Recurrent Neural Network (CRNN) - Connectionist Temporal Classification (CTC) model which results in 93.3 % accuracy. In order to deal with errors produced in the recognized text, edit distance methods are further implemented and analyzed. Damerau-Levenshtein distance method is deemed to be the most suitable, yielding a well-grounded system for medicine name recognition.**

Keywords*— *Handwriting recognition, CRNN, CTC, Fuzzy matching

I. Introduction

The prescriptions supplied by doctors become incredibly tough to interpret due to illegible and cursive handwriting [1]. Medical errors caused by unclear prescriptions are prevalent and can be detrimental to the patient, especially upon the administration of wrong drug or dosage. It has been established that even pharmacists who are responsible for dispensing medications have difficulty understanding the handwriting [2].

Various methodologies have been put forth to resolve the problem of offline handwriting recognition. Nonetheless, in most places around the world digitization of prescriptions or medical terms remains undone. Therefore, it is of utmost importance to converge the scope of these techniques to medical terms. Few deep learning approaches to tackle this task are at hand [3]. However, they are at a standstill since medical terms cannot afford to have any margin of error. A system that recognizes medical terms as accurately as possible to mitigate the problems that persist today is a necessity.

In this paper, we propose a method that applies fuzzy matching algorithms to the deep learning recognition model in order to enhance its results thereby establishing a strongly built system. The strength of deep learning for recognition is leveraged to be further improved by the string-matching algorithms and correct the wrongly predicted outputs. This system as a whole can be a stride in the recognition of medical terms and further be extended to a range of applications within the medical domain.

The primary contribution of our work is the use of fuzzy matching algorithms after model recognition to enhance model performance. Secondly, the method was evaluated on a real dataset curated with the help of 4 doctors, resulting in a 93.3% model accuracy followed by the correction of wrongly predicted terms.

The rest of the paper is organised as follows: Section II includes a summary of the pertinent literature. In Section III, the proposed methodology is described. It covers dataset collection, pre-processing, model construction, and explanation of fuzzy matching methods. Section IV presents the results of the implementation, followed by concluding remarks in Section V.

II. Literature Review

The problem with offline cursive handwriting recognition has guided the development of various technologies. In [4], Convex hull algorithm and the Support Vector Machines (SVM) model are used for feature extraction and text recognition respectively, providing an accuracy of 85%. A comparative analysis of multiple handwritten recognition techniques is done in [5]. On comparison of its accuracy rate with Artificial Neural Network (ANN) and Intelligent Character Recognition (ICR), Optical Character Recognition (OCR) was found to be the best. However, other neural network architectures have emerged as quick and dependable techniques for recognition in offline recognition systems in order to achieve high efficiency.

CRNN is implemented in [6] yielding an accuracy of 95%. Three training series were performed on normal text, cursive text and doctor's prescriptions. Recognition of lines with unconstrained text with the use of a novel connectionist system is explored in [7]. Bidirectional Long Short-Term Memory (BiLSTM) plays a very important role in entire architecture. Language model and dictionary is followed by output of CTC model to obtain final sequence.

The use of different image segmentation methods is analyzed in [8] and coupled with ANNs for recognition. On experimenting with various cursive datasets, it is found that handwriting styles, feature extraction and pre-processing methods determine the efficiency of ANNs. A series of powerful segmentation and recognition algorithms for handwriting recognition are suggested in [9]. A method of character segmentation is followed by parameter estimation such as character stroke width, height. Finally, Hidden Markov Model (HMM) and Lexicon information is combined for recognition of words. Overall efficiency corresponding to the word level recognition was improved by the utilization of trie data structure. Utilization of geometrical character analysis algorithms for handwriting recognition is explored in [10]. Center of mass of the character is used as a reference character for creation of contours. This method is designed for

978-1-6654-9291-1/22/$31.00 ©2022 IEEE

recognition of isolated characters. A study that focuses on localization of text on prescriptions [11] and further classifying it into printed and handwritten text. The images it works with are unlabeled in terms of medicine names and are not ideal for recognition.

Data Augmentation is shown to elevate the accuracy of the recognition models. A corpus of 17, 431 words containing English and Bangla words was subjected to data augmentation in [12]. On applying LSTM, the obtained accuracy was 89.5% which was 16.1% higher compared to the before augmentation scenario. A new data augmentation method [13] was designed for sequence-like characters that jointly optimizes the recognition model and augmentation. Samples were generated through an automatic learning process and proved to be helpful for the model training.

The inconsistencies in the predicted text provided scope for post processing functionalities. A correction framework is provided in [14] for the OCR output. It deals with removal of extra spaces and masking of incorrect words followed by their prediction. Upon usage of this post processing technique on the Mining Biodiversity (MiBio) dataset the average word error rate and character error rate were reduced by 5.709 and 2.07. An information extraction engine used in [15] generates text relevant to the recognized text among which the best candidate is picked by the text restoration engine. This led to an increase in accuracy by 5%.

Any application which deals with names like this are subjected to errors which are not avoidable. A method named syntactic string matching and parsing is described in [16] which uses a fuzzy matching mechanism to address such problems. But the thinning algorithm involved in preprocessing introduces barbs which derails the classification. On depicting the significance of name matching, [17] describes different name matching algorithms and provides their implementation resulting in improved data accuracy. A step forward to complexity analysis and comparison is provided in [18] for the Levenshtein, Jaro-Winkler, Soundex, N-grams and Mahalanobis distance methods. However, extensive work of their usage in this specific use case is yet to be done.

III. METHODOLOGY

This system demonstrates a solution to solve the Doctor's Handwriting Recognition problem. Visual representation of the system is shown in Fig. 1. Data is collected from 4 doctors to test the performance of the system in real life scenarios. On the collected data, preprocessing is performed followed by some augmentation methods. These images are directed to the CRNN- CTC model. The recognized text from the model is subjected to a fuzzy matching mechanism. The resulting output is the closest match found within a corpus that contains 3, 300 medicine names.

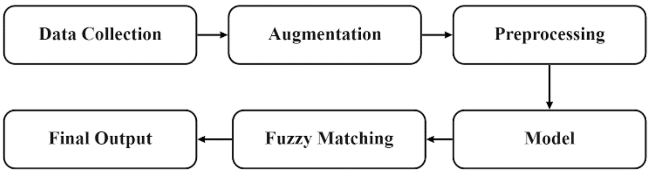

Fig. 1. The flow of execution

A. Dataset Collection

A dataset containing 600 images was compiled. The names of 50 common medicines each written by 4 doctors were collected to form 200 images. Linear stretch and perspective augmentation methods were performed on these images which resulted in a total of 600 images as mentioned in Table 1.

TABLE 1

Image Type	No. of Images
Original	200
Linear Stretch	200
Perspective Transformation	200
Total	**600**

In Fig. 2, sample images from the compiled dataset are shown.

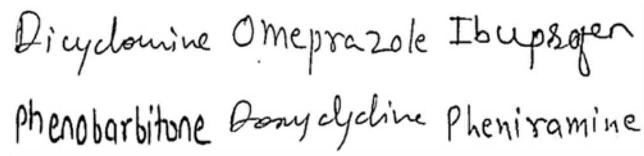

Fig. 2. Six sample Images from dataset

A separate text corpus containing 3, 300 images was curated to facilitate the fuzzy matching mechanism. A list of essential medicines was taken from the World Health Organization (WHO) website and other sources.

B. Implementation details

In this section, the proposed method is described in detail. It illustrates 4 modules namely i) Data Augmentation ii) Preprocessing iii) Model Building iv) Fuzzy matching.

I) Data Augmentation

The problem of data scarcity in this system is handled by the data augmentation techniques named stretching and perspective transformation. These data enhancement techniques are inspired from augmentation methods mentioned in Paddle OCR. The TLA algorithm is useful while dealing with sequences of characters.

After extraction of image height and width a threshold value is calculated which depends on the value of cut provided.

$$cut\ = img\_width\ //segment$$

$$thresh\ = cut\ \times 4//5$$

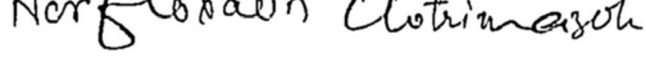

Fig. 3. Stretch Augmentation

Value of the cut after a few trials was decided as 4. Sample stretched images which transformed after a few more operations are shown in Fig. 3.

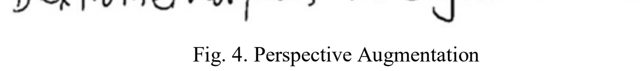

Fig. 4. Perspective Augmentation

Perspective augmentation method helps in producing new images from different views. Sample images after this transformation is completed are shown in Fig. 4.

II) Preprocessing

Each image is cropped to 100 x 400 pixels size. The image is then converted to grayscale after loading. Gaussian blurring as shown in Fig. 5, is performed on the image to filter out the gaussian noise generated in images during acquisition.

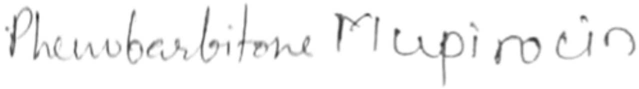

Fig. 5. Blurring

The next operation was binarizing the image with adaptive thresholding. Other global thresholding methods such as Otsu thresholding fail in images with varying illumination and shadowing as only one threshold value is used. Adaptive thresholding computes the threshold value for various neighborhoods and provides better segmentation results It also eliminates the need for using complex and time-consuming neural network architectures.

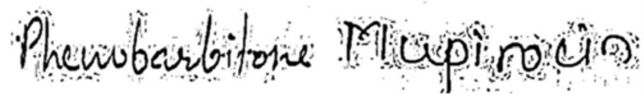

Fig. 6. Thresholding

The output of single adaptive thresholding is shown in Fig. 6 and its application on a gaussian blurred image is shown in Fig. 7, which is the input to the recognition model.

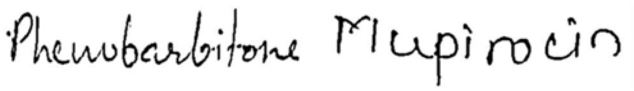

Fig. 7. Blurring and thresholding

III) Model Building

Visual feature extraction is carried out by using a convolutional neural network (CNN) which is part of the CRNN model as illustrated in Fig. 8. The architecture consists of a stack of 2 convolutional layers consisting of a 3 x 3 kernel. Nonlinearity is introduced by the use of Rectified Linear Unit (ReLU) activation. The weights of the convolution layers are initialized to he-normal. Max pooling with 2 strides is applied after each convolutional layer. A Dense layer followed by the reshaping layer is used to reduce the number of parameters used for training.

The output is fed to the 2 BiLSTM layers. The dropout rate for the training is set to 0.25 in each of the LSTM layers. The last layer employs the softmax activation function to determine which output class has the highest probability inside a frame. This output is fed to the CTC decoder. As the input image varies in nature, the CTC loss is specifically created to optimize both the length and the classes of the projected sequence, unlike other loss functions that only optimize a single objective function. Within the back-propagation process, concurrent training of the LSTM weights and convolution filters is carried out. Adam optimizer is utilized for the training with a learning rate of 0.001.

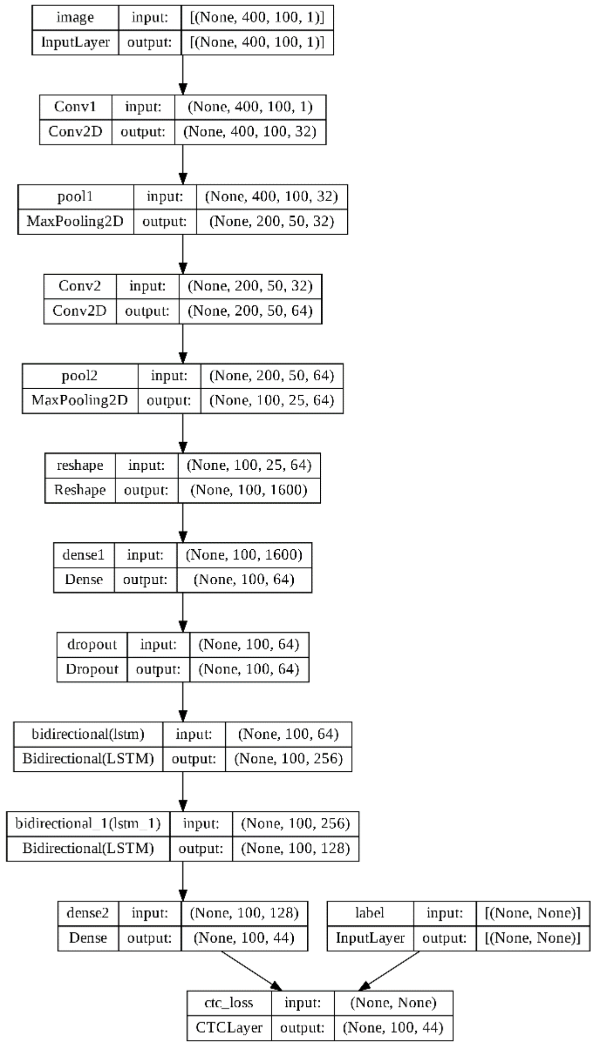

Fig. 8. CRNN- CTC Model Summary

IV) Fuzzy Matching

A technique useful for identification of two elements of text, strings, or entries which are approximately similar but are not the same is Fuzzy Matching. Since the output of the recognition model contains text approximately equal to the desired text, this mechanism is applied post recognition. Various distance based fuzzy matching algorithms were implemented and compared.

i. Levenshtein distance (LD):

It is an algorithm used to compare and compute the distances of two different phonetic strings. Minimum number of single character changes such as additions, replacements, or deletions necessary to transform one word into another are calculated. There are multiple ways of implementation of this operation. Recursive approach results in time complexity of O(3N), where n is the length of the longest string. Whereas the time complexity of Dynamic Programming results in O(MN) where the lengths of the strings are M and N.

ii. Damerau–Levenshtein distance (DLD):

An edit distance metric where transposition operation is allowed with three single character edit operations. Upon its implementation on the predicted words and in accordance with the text corpus of 3, 300 medicines, a final output with significant improvement is obtained. Implementation in this system is based on an optimal string alignment distance algorithm. Time complexity required for this method is O(NM) having O(M) space where M is the length of the corpus and N is the length of the predicted sequence.

iii. Cosine Similarity:

This is a similarity measure which utilises the cosine of the angle between two non-zero vectors in an inner product space. Value of cosine similarity lies in the positive space which is bounded in [0,1]. Time complexity of this similarity measure is quadratic i.e., O(N2). Use of this similarity measure in this application however didn't improve model predictions.

iv. Hamming distance:

It is another edit distance method and is given by the number of places in two strings where the symbols differ. But the Hamming distance is applicable only to strings having equal length and it utilises substitution. Hence, it is unsuitable for this application.

IV. Results and Discussions

The CRNN model was trained over 540 images with a validation set of 20 images. After training for 200 epochs the observed training and validation losses were 0.6536 and 0.2513 respectively. 40 images were used for testing which produced an accuracy of 93.33%

To improve the wrong predictions given by the model a fuzzy matching block was added that outputs the correct word. The distance-based algorithms implemented and compared under this were Levenshtein, Damerau-Levenshtein, and Hamming distance. The incorrect predictions produced by the model and the corrected predictions made by these fuzzy matching algorithms are shown in Table 2.

TABLE II

Model Predictions	DLD, LD	Cosine	Hamming
Amoxicdllin	Amoxicillin	Sodium Fluoride	Sodium Fluoride
Hydrocortsone	Hydrocortisone	Sodium Fluoride	Sodium Fluoride

The accuracy percentages of all the methods are illustrated in Fig. 9. The orange part represents the percentage of correct predictions while the gray part corresponds to the incorrect predictions.

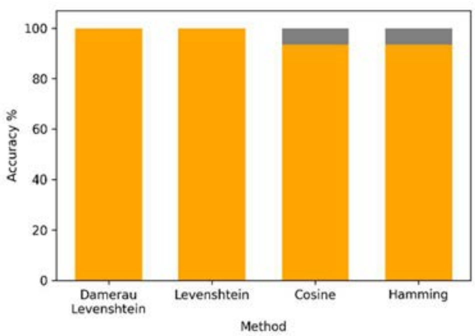

Fig. 9. Comparison of methods based on accuracy

Comparative study conducted in terms of accuracy, time consumption and space consumption are shown in the table given below. The Levenshtein and Damerau-Levenshtein distance methods lay out 100% accuracies each, whereas Hamming distance provides an accuracy of 93.33. While some aforementioned methods provide a good accuracy they differ in terms of time as shown in Table 3.

TABLE III

Method	Accuracy	Time consumption (seconds)	Space consumption (MB)
Damerau - Levenshtein	100	1.067	0.096
Levenshtein	100	11.31	0.112
Cosine	93.33	2.464	0.239
Hamming	93.33	3.083	0.260

While the Levenshtein method has better space consumption, its time consumption is visibly higher (11.31s). Its variation, the Damerau- Levenshtein method has significantly less space consumption over all other methods (0.096 MB) along with low time consumption (1.067s). Overall, Damerau- Levenshtein works most efficiently in all the performance measures.

V. Conclusion

Handwriting recognition is inherently an ambiguous problem and doctors handwriting in particular is considered complex and illegible most of the time. Our system provides a solution for its recognition as well as a novel addition of string-matching algorithms to improve the accuracy of model predictions. 93.33% is the highest testing accuracy which is given by the model. Further combined with string matching algorithms, the wrong predictions are corrected to give 100% accuracy in the end stage.

In order to rise in terms of scalability, more medical images will be integrated through multiple doctors to form a larger medical term dataset. Along with this, localisation, and recognition of text in the entire prescription will be implemented as a step forward to digitization of medical records. The proposed system will help in reducing medical errors and thus solve the problems created by failure of deciphering doctors handwriting.

References

[1] E. Hassan, H. Tarek, M. Hazem, S. Bahnacy, L. Shaheen and W. H. Elashmwai, "Medical Prescription Recognition using Machine Learning," 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), 2021, pp. 0973-0979

[2] 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

[3] 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

[4] 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), 2017, pp. 1-5

[5] Tiwari, U., Jain, M., Mehfuz, S. (2019). Handwritten Character Recognition—An Analysis. In: Singh, S., Wen, F., Jain, M. (eds) Advances in System Optimization and Control. Lecture Notes in Electrical Engineering, vol 509. Springer, Singapore.

[6] R. Achkar, K. Ghayad, R. Haidar, S. Saleh and R. Al Hajj, "Medical Handwritten Prescription Recognition Using CRNN," 2019

International Conference on Computer, Information and Telecommunication Systems (CITS), 2019, pp. 1-5

[7] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke and J. Schmidhuber, "A Novel Connectionist System for Unconstrained Handwriting Recognition," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 5, pp. 855-868, May 2009

[8] U. Choudhary, S. Bhosale, S. Bhise and P. Chilveri, "A survey: Cursive handwriting recognition techniques," 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017, pp. 1712-1716

[9] N. Arica and F. T. Yarman-Vural, "Optical character recognition for cursive handwriting," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 6, pp. 801-813, June 2002

[10] Kacalak, W., Majewski, M. (2012). Effective Handwriting Recognition System Using Geometrical Character Analysis Algorithms. In: Huang, T., Zeng, Z., Li, C., Leung, C.S. (eds) Neural Information Processing. ICONIP 2012. Lecture Notes in Computer Science, vol 7666. Springer, Berlin, Heidelberg.

[11] Dhar, Dibyajyoti, Avishek Garain, Pawan Kumar Singh and Ram Sarkar. "HP_DocPres: a method for classifying printed and handwritten texts in doctor's prescription." Multim. Tools Appl. 80 (2021): 9779-9812.

[12] 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

[13] Luo, Canjie, Yuanzhi Zhu, Lianwen Jin and Yongpan Wang. "Learn to Augment: Joint Data Augmentation and Network Optimization for Text Recognition." 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020): 13743-13752.

[14] S. Karthikeyan, A. G. S. de Herrera, F. Doctor and A. Mirza, "An OCR Post-Correction Approach Using Deep Learning for Processing Medical Reports," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 32, no. 5, pp. 2574-2581, May 2022

[15] Q. Chen, T. Gong, L. Li, C. L. Tan and B. C. Pang, "A Medical Knowledge Based Postprocessing Approach for Doctor's Handwriting Recognition," 2010 12th International Conference on Frontiers in Handwriting Recognition, 2010, pp. 45-50

[16] Suresh, R.M., Arumugam, S. (2006). Fuzzy Technique Based Recognition of Handwritten Characters. In: Di Gesú, V., Masulli, F., Petrosino, A. (eds) Fuzzy Logic and Applications. WILF 2003. Lecture Notes in Computer Science(), vol 2955. Springer, Berlin, Heidelberg

[17] A. Pilania and G. M. M. Kumaran, "Comparative Study of Name Matching Algorithms," 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom), 2019, pp. 1174-1178

[18] H. A. Maarif, R. Akmeliawati, Z. Z. Htike and T. S. Gunawan, "Complexity Algorithm Analysis for Edit Distance," 2014 International Conference on Computer and Communication Engineering, 2014, pp. 135-137