A Survey on Recurrent Neural Network and Various Techniques for Handwritten Recognition

Geetanjali Bhagwani[1] Ompriya Kale[2]
[1]M.E Student[2]Assistant Professor
[1,2]Department of Computer Engineering
[1,2]L.J. Institute of Engineering & Technology, Ahmedabad, Gujarat, India

Abstract— At present large number of manuscripts, books, journals, and articles remain largely inaccessible in library archives. Keyword spotting refers to the process of retrieving all instances of a given keyword from these documents. In the present paper, a novel keyword spotting method for handwritten documents is0020obtained using different various systems for unconstrained handwritten recognition. A new technique is used for robust keyword spotting that uses bidirectional Long Short-Term Memory (BLSTM) recurrent neural nets and CTC Token Passing Algorithm to incorporate contextual information in documents. Each surveyed method briefly discusses the keyword searching issues and solutions.

Key words: CTC Token Passing Algorithm, Recurrent Neural Network, Handwritten Recognition

I. INTRODUCTION

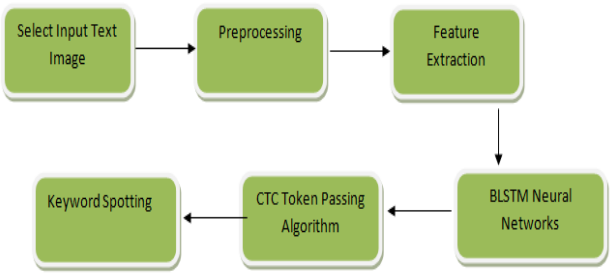
The goal of keyword spotting is to reliably detect the instant of a specific word in a given document. The document can be a printed text or handwritten text. Searching or browsing handwritten text is in high demand and necessary. Without transcribing the data, a user should still be able to search for any possible word, just like using a search engine. There are large number of historical manuscripts, articles, letters, journals, books, and notes that remain largely inaccessible. All these document requires enough accuracy to support for traditional text searching. The following are the techniques designed for the task of keyword spotting.

- Word based Keyword Spotting: The task of keyword spotting as detecting a word or a phrase in an image has been initially proposed for printed text and a few years later in for handwritten text.
- Line based Keyword Spotting: All former approaches require the text in an image of a line which is to be segmented into individual words before keyword spotting takes place.
- Document based Keyword Spotting: To work on completely un-segmented pages of text, a system can either include either a segmentation approach or a non-segmentation approach on the documents.

Neural Networks (NN) have found their way into keyword spotting with so called Bidirectional Long Short-Term Memory (BLSTM) recurrent NN [6].

In this work, we have used BLSTM NN which has two parts for handwritten recognition. The BLSTM neural network is bidirectional, meaning the text line is processed from both left-to-right and right-to-left. This is done because context from both sides of a character is useful to improve the recognition. The information from two separate input layers is collected in two separate LSTM layers, respectively, and finally joined in the output layer. The output layer contains one node for each possible character as well as one additional node, which is activated when no evidence about the presence of any character can be inferred.

The modules that are included in this work are as follows:



A. *Preprocessing*

The first one is a performed by the neural network in which all the documents are already segmented into individual text lines. It means that the input word images are been normalized.

B. *Feature Extraction*

A sequence of feature vectors is extracted from each line, which is forwarded to the neural network. A sliding window is moved over the image in all the four directions and the result is extracted. The process maps each position of an input sequence to a vector, indicating the probability of each character at that position.

C. *CTC Token Passing Algorithm*

This algorithm takes the sequence of letter probabilities from the neural network as its input and computes a likely sequence of words of ASCII characters. The best path result through the letter probability sequence is computed that correspond with the letters from the input word. The value of that path is then returned as a matching score. During the following loop over all input sequence positions and the tokens are updated, so that a) the token's corresponding letter occurs at some position, b) in the best path result, all letters of the word occur in the proper order, c) between two successive letters, only node activations are considered and d) if two successive letters of the word are same, then at least a node must exist between them. A set table is created in which all valid tokens are stored that can act as predecessor to the token according to the constraints mentioned above. Then, the probability of the best token in table is multiplied to obtain the result.

Lastly the output obtained is the keyword spotted on the input data document. The returned probability of a word still depends upon the word's length. We take the logarithm of the probability and divide it by the search word's length to receive a normalized value which can then be threshold. An approximation of the keyword's length that works very well is to use the number of characters of the word. This value is constant throughout the test set for each keyword.

II. LITERATURE REVIEW

The following are the survey done on various methods for handwritten recognition:

A. Word Spotting: A New Approach To Indexing Handwriting [1]

The present paper deals with the word recognition by word based keyword spotting approach and the word images are matched against each other. The matching phase is expected to be the most difficult part of the problem. This is because unlike printed text, there are variations even in a single person's handwriting which is difficult to model.

The matching is done in a number of phases. Pruning is done on the basis of the area of the word images and the width or height ratios of the word images. Then, in matching phase two algorithms are used: (EDM) Euclidean distance mapping and (SLH) Scott and Longuet Higgins algorithm.

EDM matches words assuming that the transformation between the words may be modeled by a translation (shift). First the images are aligned in both the direction. In the vertical direction, it is aligned by the baselines of the two images. The baselines are declared at which there is maximum number of white pixels. The images are aligned by making their left hand sides intersection in the horizontal direction. From this, XOR image is computed. Then EDM is obtained from this XOR image by assigning white and black pixel to the image. To get the accurate result and obtain minimum error rate last two stages are repeated.

SLH matches words assuming that the transformation between the words may be modeled by an affine transform. This algorithm overcomes the drawback of EDM when there is significant distortion in the words. An affine transform is a linear transformation between coordinate systems. The two criteria that restrict the choice of algorithm are grey matching technique and correspondence between images and affine transform. Though, the affine matching algorithm is much more accurate than the Euclidean distance mapping technique.

Here, the examples include the W. B. DuBois collection at the University of Massachusetts and the early Presidential libraries at the Library of Congress. Experiments are shown demonstrating the feasibility of the approach for indexing handwriting.

The EDM does not account for any distortions and thus performs poorly when the handwriting is bad. The SLH algorithm produces the correct rankings almost always - this is true even if the handwriting is bad. The work provides feasibility only if the words are written by single author.

B. Hmm-Based Handwritten Symbol Recognition Using On-Line And Off-Line Features [2]

This paper features out the solution for recognizing on-line sampled handwritten symbols. Online data is a sequence of strokes obtained during writing; each stroke itself is a sequence of (x.y) coordinates corresponding to the pen positions. Within the proposed symbol recognition system based on Hidden Markov Models different kinds of feature extraction algorithms are used analyzing on-line features as well as off-line features and combining the classification results.

One of these algorithms is a typical on-line algorithm using the temporal information of the handwriting for generating parameters. The remaining two algorithms are offline algorithms generating parameters by extracting from the image of the symbol.

Preprocessing is done on the on-line data before applying extraction algorithm. In the on-line system, each hidden stroke represents the pen movement from the final pen position of the actual stroke to the starting position of the successive stroke. In the off-line system, the result is based on the image of the handwritten symbol is calculated by interpolating the on-line sampled data.

Within the system total 84-character alphabet are used which includes three symbols like 'Dot',

'Minus' and 'Fraction' and remaining other contain upper and lower case letters as well as digits, mathematical operators and other special symbols are used for HMM training and experiments.

By conducting writer-dependent recognition approach, it is observed that the recognition rates as well as the reliability of the results are improved by using the proposed recognition system. Also, by applying handwriting data not representing symbols out of the given alphabet, rejection rate has increased

C. Generating Text with Recurrent Neural Network [3]

The main goal of this paper to demonstrate the power of multiplicative Recurrent Neural Networks (RNN) trained with the new Hessian-Free (HF) optimizer by applying them to the task of predicting the next character in a stream of text. In reality, the relationship between the data parameters and the dynamics of the RNN is highly unstable which makes gradient descent ineffective. Thus two ways are dealt to overcome this ineffectiveness: Long-Short Term Memory and Echo State Network.

Multiplicative RNN was used because from viewing an RNN as a model of an unbounded tree in which each node is a hidden state vector and each edge is labeled by a character that determines how the parent node gives rise to the child node. RNN was much more powerful than the Markov model because the distributed representation of a node allows different nodes to share knowledge.

In this work it is observed that the MRNN, features out high-quality character-level language models to predict the next character in a sequence, when trained with HF. It is observed that RNN is much more powerful then HMM.

In the experiment the datasets used are a long string of characters from an 86-character alphabet that includes digits and punctuation, together with a special symbol which indicates that the character in the original text. Examples are English Wikipedia, articles from New York Times and a corpus of machine learning papers. This is the largest RNN application used till date.

D. One Dimensional Representation Of Two Dimensional Information For Hmm Based Handwritten Recognition [4]

The most powerful character recognition system Hidden Markov Model (HMM) is used as recognizers in this paper. There are many studies, which employ hundreds of features on various types of HMM, which can be found in [7, 8, 9].

The extension of the HMM into two-dimensional image processing applications are not as successful as the one-dimensional cases. This is the reason, a new set of one-

dimensional discrete is introduced here, constant length features to represent 2D shape information for HMM, based handwritten OCR issue.

The system employs the gray scale and binary information in a mixed way to extract the maximal amount of information for both segmentation and recognition stages on offline handwritten recognition. **A** two-step segmentation algorithm, proposed by the authors of this paper [10]**,** finds the nonlinear character segmentation paths in free style handwriting.

Recognition is accomplished in two stages: first, a new set of features is extracted from the segmented regions, which mostly corresponds to a character. Training stage of HMM involves dynamical adjustments of the parameters of the normalized feature set. Finally, a recognition based segmentation algorithm resolves handwriting strings by maximizing a cumulative information measure, computed from the probabilities.

The dataset tested in the experiment is a local database with 4000 handwritten connected digits, collected from 20 persons of which 50 samples were used. The normalization parameters are estimated by gradually increasing the parameters and calculating the corresponding recognition rates. Another dataset tested on 10000 isolated digits randomly selected from NIST Special Database 1 and Special Database 7, of which 100 samples were used. In this recognition rate increases to 97%.

E. Indexing Of Handwritten Historical Documents–Recent Progress [5]

Searching and indexing historical handwritten collections is a very challenging problem. An approach of word spotting is used in here which use image matching for calculating pair wise distances between the word images to find similarity.

Here matching techniques are categories in two ways: image matching approaches that compare images pixel-by-pixel, and feature-oriented techniques, that extract image features.

Image matching is investigated by three ways: XOR[11], Sum of Squared Differences SSD[11] and Euclidean Distance Mapping EDM[11].

Feature based techniques are also investigated in three ways: Scott and Longuet-Higgins algorithm [14]SLH, Shape Context matching SC [12], Dynamic Time Warping algorithm DTW[12], correlation technique CORR[13]. DTW and CORR are discussed in detail in the paper with the experiment.

The dataset tested in this paper is a set of 100 page images of the manuscript of George Washington with well over 6000 pages. Challenges remain, including the creation of word clusters and the necessity of speeding up all these algorithms sufficiently, so that large collections can be handled in a reasonable amount of time.

III. CONCLUSION

As compared to other literatures the BLSTM NN uses line based approach then word based approach for keyword spotting. Although the system needs training, it does not require bounding boxes around characters or words as often needed in the keyword spotting literature.

REFERENCES

[1] R. Manmatha, Chengfeng Han and E. M. Riseman, Word Spotting: A New Approach to Indexing Handwriting, IEEE Computer Security Conference on Computer Vision and Pattern Recognition, 631 – 637, 1996.
[2] Hans-Jiirgen Winkler: HMM-Based Handwritten symbol recognition using on-line and off-line features, IEEE International Conference on Acoustics, Speech, and Signal Processing, 3438 - 3441 volume 6, 1996.
[3] Ilya Sutskever, James Martens, Geoffrey Hinton: Generating Text with Recurrent Neural Networks, Proceedings of the 28 th International Conference on Machine Learning, Bellevue, WA, USA, 2011.
[4] Nafiz Arica, Fatos. T. Yarman-Vural: One Dimensional Representation of Two Dimensional Information for HMM Based Handwritten Recognition, IEEE International Conference on Image Processing, 948 - 952 volume 2, 1998.
[5] R. Manmatha, Toni M. Rath: Indexing of Handwritten Historical Documents - Recent Progress, in Symposium on Document Image Understanding Technology, 2003, pp. 77–85.
[6] S. Fernandez,´ A. Graves, and J. Schmidhuber, "An Application of Recurrent Neural Networks to Discriminative Keyword Spotting," in 17th Int'l Conf. on Artificial Neural Networks, ser. Lecture Notes in Computer Science, vol. 4669, 2007, pp. 220–229.
[7] S. Connel, "A Comparison of Hidden Markov Model Features for the Recognition of Cursive Handwriting", Comp. Science Dept., Michigan State Univ., MS Thesis (1996)
[8] A. Atici, F.T. Yarman-Vural, "A Heuristic Method for Arabic Character Recognition, Joum. of Signal Processing", vol. 62, issue.1, 1997.
[9] N. Arica, F. T. Yarman-Vural "HMM Based Handwriting Recognition", Proc. of the I Znd Int. Symposium on Computer and Information Sciences, 242-247 (1997).
[10] N. Arica, F. T. Yarman-Vural, " A New Scheme for Off-Line Handwritten Connected Digit Recognition" ICPR'98 (accepted).
[11] S. Kane, A. Lehman and E. Partridge: Indexing George Washington's Handwritten Manuscripts, Technical Report MM-34, Center for Intelligent Information Retrieval, University of Massachusetts Amherst, 2001.
[12] T. M. Rath and R. Manmatha: Word Image Matching Using Dynamic Time Warping, to appear in Proc. of the Computer Vision and Pattern Recognition Conf. 2003
[13] J. L. Rothfeder, S. Feng and T. M. Rath: Using Corner Feature Correspondences to Rank Word Images by Similarity, CIIR Technical Report MM-44, 2003.
[14] G. L. Scott and H. C. Longuet-Higgins: An Algorithm for Associating the Features of Two Patterns. Proc. of the Royal Society of London B224 (1991) 21-26.

[15] Volkmar Frinken, Andreas Fischer, R. Manmathay, and Horst Bunke:" A Novel Generating Text Method Based on Recurrent Neural Networks", IEEE Trans. on Pattern Analysis and Machine Intelligence, Volume: 34, Issue: 2, 211-224, 2012.

