A Review of Different Neural Network Architectures used for the Purpose of Handwriting Recognition

Prof.Gopal Jadhao[1], Abhishek Wahane[2], Khushagra Nakhate[3], Aman Ganvir[4], Nikhil Waghmare[5]

[1-5]Dept. of CSE-IT, JD College of Engineering and Management, Nagpur

---

Abstract— In recent decades, rapid increment in computational power accompanied by enormous amount of data being generated has enabled extensive use of neural networks for solving machine learning problems especially in pattern recognition tasks. This paper aims to briefly review three of the most prominent approaches namely, Convolutional Neural Network, Recurrent Neural Network and Hidden Markov Models and discuss their most notable applications for the problem of Offline Handwriting Recognition. We first give an outline of a typical Handwriting Recognition System, briefly talking about each of the steps involved and then give and overview of these three technologies while discussing a few important literatures of each.

Keywords— Offline Handwriting Recognition, Convolutional Neural Network, Recurrent Neural Network, Hidden Markov Model

1. INTRODUCTION

The problem of Handwriting recognition (HWR) is being actively researched from more than last 40 years, and while significant progress have been made, there is still a wide gap between performance of recognition systems and human capabilities, which is why it still remains an open problem. Recognition of unconstrained handwriting is a very difficult pattern recognition task due to seemingly infinite variations of writing styles of different authors. Even the writing of the same author may vary under different conditions, writing material and emotional state of the author. Also, cursive handwriting makes it more difficult to segment words into individual character making naive approaches like pre-segmenting words into characters and classifying each segment impractical.

The problem of Handwriting recognition can be classified into two types: offline recognition and online recognition. In online recognition a time series of coordinates, representing the movement of the pen-tip, is captured, while in the offline case only an image of the text is available. Thus, the problem of online recognition is relatively easier to that of offline, since it provides a wider range of inputs like the trajectory of pen well as temporal position of writing process which makes it significantly easier. In fact, the many argue that the problem of online recognition has been solved.

The focus of this paper is offline recognition. Section 1, of this paper gives a brief information about database that are most popularly used for training and evaluation of handwriting recognition systems. Section 2 list the different components in the structure of a typical handwriting recognition system. Section 3 summarizes the architectures of the three different system and review some important literature and finally Section 4 concludes gives some conclusive thoughts.

2. DATABASES

The data is one of the most important part of any machine learning model which provides the input and enables training of parameters. Some of the most popular databases used for training and benchmarking Handwriting recognition systems are listed below:

A. *IAM Database*

The IAM database consists of images of handwritten documents. They correspond to English texts extracted from the LOB corpus, copied by different writers. The database is split into 747 images for training, 116 for validation, and 336 for evaluation. We used a 3-gram language model limited to the 50k most frequent words from the training set. It was trained on the LOB, Brown and Wellington corpora. The passages of the LOB corpus appearing in the validation and evaluation sets were removed prior to LM training. The resulting model has a perplexity of 298 and OOV rate of 4.3% on the validation set (329 and 3.7% on the evaluation set) [16].

B. *RIMS Database*

The Rimes database consists of images of handwritten paragraphs from simulated French mail. The setup for the ICDAR 2011 competition is a training set of 1,500 images, and an evaluation set of 100 images. We held out the last 149 images from the training set for system validation. We built a 4-gram language model (LM) with modified Kneser-Ney discounting from the training annotations. The vocabulary is made of 12k words. The language model has a perplexity of 18 and out-of-vocabulary (OOV) rate of 2.9% on the validation set (18 and 2.6% on the evaluation set) [16].

C. *C-Cube Database*

The C-Cube is a public database available for download on the Cursive Character Challenge website (http://ccc.idiap.ch). The database consists of 57,293 files, including uppercase and lowercase letters, manually extracted from the CEDAR and United States Post Service (USPS) databases. All images are binary and with variable size. The data are unbalanced and there is a big difference in the number of patterns among the letters .

D. IFN/ENIT Database

The IFN/ENIT database represents a standardized set of handwritten Arabic town/village names [96]. It consists of scanned forms of more than 400 writers with about 26,400 city names containing 210k+ characters. In addition to the images and their annotation, further information as for example the correct baseline of the cropped and preprocessed words is also provided.

3. OVERVIEW OF HWR SYSTEM

E. Preprocessing

The input data for a handwriting recognition system generally consists of 2-dimensional grayscale images which may need some pre-processing. Pre-processing is the first unit of a Handwriting recognition system which aims improve the quality of the handwriting by reducing noise and some variations in the image which do not affect the identity of the word and transform it into a form better suited for later steps. The amount of effort that needs to be invested into the pre-processing depends on the given data, for example, poor image acquisition may require some additional image enhancement techniques. The grayscale images may be converted into black and white by means of binarization. Some modern databases already implement most of the pre-processing.

F. Normalization

The writing styles of different authors and slight flaw in acquisition of images, further add another dimension of undesirable variety to the data. Normalization attempts to remove writer-specific characteristics of the handwriting to make writings from different authors looking more similar to each other. Some of the most common characteristics are:

- Slope (the angle between the x-axis and the implicit horizontal axis on which the word is aligned)
- Slant (the angle between the y-axis and the direction of the vertical strokes)
- Stroke width
- Size

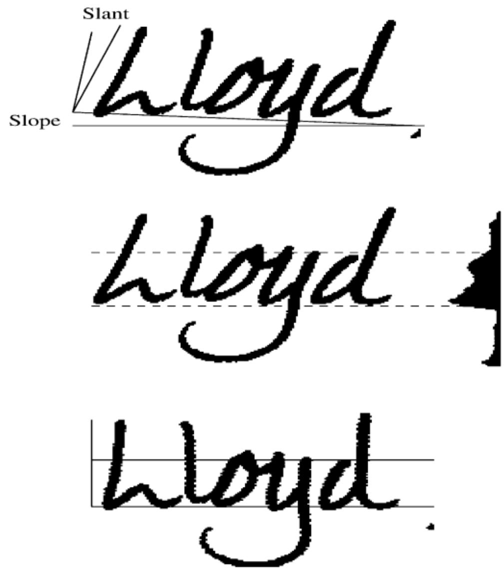

Figure 1 The word before normalization shows slope and slant different that 0. After normalization, the word appears horizontal with ascenders and descenders aligned with the vertical axis [22]

Removal of Slope and Slant can be performed using various techniques most of which start by estimation of baseline and top minuscule line [1] Most of the slope and slant correction techniques are inspired by the method proposed in Ref [2].

G. Segmentation

Segmentation involves dividing an image into local fragments which give some basic spatial information for recognition process. However, segmentation is difficult for cursive or unconstrained text, unless the words have already been recognized. This creates a circular dependency between segmentation and recognition that is often referred to as Sayre's paradox [11]. One approach to avoid this is to segment the text into basic strokes rather than characters. The stroke boundaries can be defined in terms of minima of velocity, minima of the y-coordinates or the points of minimum curvature. Another solution to this problem is implemented by Hidden Markov Models which integrate segmentation and recognition in the same step.

H. Feature Exraction

Mainstream feature extraction can be broadly classified into two type: handcrafted and sliding window approach. Handcrafted features are manually designed features to exhibit the word shape structure such as edges, lines and curves while the other approach involves learning from pixel features in which a sliding window moves from left to right scanning for relevant features that might be helpful in recognition.

Recent alternative to handcrafted features include using convolutional neural network to learn features. Another alternative is to use dimensionality reduction using Principle Component Analysis which is among the most popular unsupervised feature extraction methods. Supervised alternatives include where the recognition has direct influence on the extraction process which is typically the case with Multi-Layer Perceptron (MLP) neural networks.

I. Recognition

The recognition process consists of predicting the character with highest compatibility with the handwritten input image, given a trained character model and a statistical language model of the possible character or word sequence.

The use of a word lexicon during recognition is optional, and its use generally results in a lower error rate. The lexicon is estimated from a suitably large text corpus. The language model, which provides the probability of any character or word sequence, is also estimated from the same corpus[19].

In the next section three well established approaches for recognition are briefly reviewed.

4. RECOGNITION MODEL ARCHITECTURES

J. Convolution Neural Network

Convolutional Neural Network is a neural network architecture that is widely used for analyzing image data.

Convolutional Networks integrate three architectural concepts: 1) local receptive fields; 2) shared weights (or weight replication); and 3) spatial or temporal subsampling to preserve some degree of shift scale and distortion invariance.

Normalized images of the characters are fed into the input plane of the network. Each unit (neuron) in the subsequent layer receives input from a set of units located neighborhood in the previous layer. The connecting receptive fields on the input is somewhat analogous to the locally sensitive, orientation-selective neuron's in visual system of brains of mammals. These local receptive fields fire up activation of neurons corresponding to elementary visual features such as edges, endpoints, corners, curves. These constitute the low-level features and are then combined by subsequent layer to extract higher order features. Additionally, elementary feature detectors are useful across entire image and act as "filter" which detect the presence of features and their position. These units are organized in a plane and the set of output of units in such a plane constitutes what is called a feature map. A convolution layer consists of several such feature maps to extract multiple features at each location. [3]

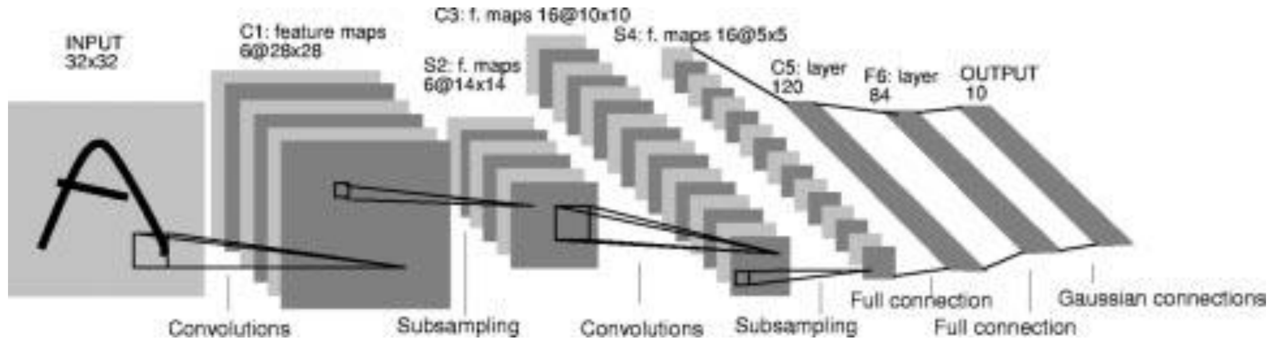

Figure 2. Architecture of LeNet-5, a typical convolutional neural network used for character recognition [3]

Fig. 2 shows LeNet-5, a convolutional neural network for recognizing characters. It consists of several layers where each convolution layer is followed by a sub-sampling layer which reduces the dimension of corresponding convolution layer's feature may by a factor of 2.[4] The input layer receives 32x32 grayscale images. The units in the first hidden layer are organized in six planes, each of which is a feature map having 25 weights, connected to a 5x5 area in the input, called the receptive field of the unit. Since receptive fields of a feature map has contiguous units connected to the previous layer, neighboring fields overlap. Since all units in a feature map share same set of weights and biases, they detect the same features at all possible locations of input. Whereas, different feature maps have different set of weights and biases resulting in detection of different types of local features. Thus, layer one extracts six different types of features by six units in identical locations in six feature maps and store the states of these units in corresponding locations in a feature map. The following sub-sampling (S4) layer reduces the size of the layer to 14x14. The third layer (C3) is a convolutional layer consisting of 16 different feature maps of size 10x10 followed by a sub-sampling-layer consisting of 16 feature maps size of size 5x5 which reduces the output of previous layer (C3) to size 5x5.The next convolutional layer (C5) consists of 120 feature maps of size 1x1 resulting in full connection of layer S4 and C5. Finally, C5 connects to fully connected layer F6 containing 84 units which consists of 10,164 trainable parameters. The final output layer of the network consists of 84 Euclidean RBF units, one for each class. Ref [3] gives a more comprehensive insight of Convolutional networks along with architectural details and working of LeNet-5.

The selective activation of units in the receptive fields helps retain spatial information of the image, which is not possible in case of fully connected deep neural network, since all the spatial information is completely lost while converting the 2-dimensional image into one dimensional vector which is needed by the input layer of the network. The retention of information in the neighboring pixels in the image helps in identifying local features which enable Convolutional Neural Networks to recognize visual patterns in an image. This merit has made CNN a state of the art for general computer vision tasks.

Convolutional Neural Networks have been applied to the problem of handwriting recognition and have been met with encouraging results. CNNs hold current state of the art for performance on the MNIST database achieving error rate of 0.17 percent [10]. For general unconstrained handwriting recognition CNN widely used in combination with different models. Some of the most prominent application to HWR are as follows:

In Ref [6] Th´eodore Bluche et al., show that using recognition systems with convolutional neural network for feature extraction yield better results than using traditional approaches for feature extraction such as handcrafted features or PCA on pixels. Their experiment was performed by using Hidden Markov model in Hybrid model with convolutional neural network on Rimes database. Their architecture of CNN is similar to that of LeNet-5 consisting of three convolutional layers with 5×5px filters, followed by sub-sampling operations. On top of this architecture is a fully connected hidden layer, and an output SoftMax layer with as many units as there are states in the HMM models (490). They combine CNN with Hidden Markov model using two methods. The first one involves Hybrid combination and other one involves tandem scheme. They have used two methods for feature extraction which are Grapheme segmentation and sliding widow approach. While comparing all six of the proposed systems, they have noted that the grapheme systems are globally worse than the systems based on sliding windows. They have showed that showed that a hybrid ConvNN/HMM is better than a GMM-HMM based on handcrafted features, and comparable to a hybrid MLP/HMM (also based on these features) in the case of explicit over segmentation into graphemes. They also conclude that the sliding window and grapheme approaches are complementary: the best combinations of the presented system always include a system based on graphemes.

Dewi Suryani et al. in Ref [7], show that using convolution neural networks for initializing filters yields better results than randomly initializing filters. Their approach involves combining Convolutional Neural Network and Bi-directional Long Short-Term Memory (BLSTM) architecture of recurrent neural networks with hybrid Hidden Markov Model framework. Their experiment was conducted on openly available offline Chinese handwriting database called CASIA consisting of isolated characters (HWDB 1.0- 1.2) and handwritten text (HWDB 2.0-2.2) with 7356 classes in total for training. To evaluate their model, they also used the handwritten text dataset from the ICDAR 2013 Handwritten Chinese Character Recognition Competition. Their proposed system has CNNs composed of 18 convolutional layers and 3 max pooling layers with filter sizes of 1 × 1, 3 × 3, and 5 × 5 and 32 or 64 feature maps. The convolutional layers with 1 × 1 filters are used to reduce the number of parameters by reducing the feature maps. The remaining of the max pooling layers use pooling size 2 × 2. The fully connected layer is replaced by three BLSTM layers containing 512 memory cells for each direction. They achieved 92% CR (Correct Rate) on HWDB 1.1 and 92.20% on the ICDAR 2013 dataset in their preliminary experiment. While in their main experiment, they have got 83.50% accuracy on evaluation set which is comparable to 89.4% the state of the art achieved by MDLSTM + CTC which is much more complex system in comparison.

In Ref [8], P. Y. Simard, D. Steinkraus and J. C. Platt describe a couple of practices that can be applied to visual document analysis task to yield better results. First, they propose introduction of general set of elastic distortions that vastly expanded the size of the training set. This can be done by applying transformations to generate additional data and let the learning algorithm infer the transformation invariance. To augment the size of training data, simple distortions such as translation, rotations, and skewing can be used by applying affine displacement fields to images Second, they present a new method for implementing convolutional neural networks that is much easier than previous techniques and also easy to debug. They proposed using simpler loops for convolution and use modular debugging. Using these techniques, they achieved best performance on MNIST database at the time.

K. Recurrent Neural Network

Most feed-forward neural networks are incapable of handling sequential data because they assume independence of data points. This is why recurrent neural networks outperform feed-forward networks as they are able to the output of previous time step in their internal state allowing them to make use of past context. Recurrent neural networks (RNNs) are connectionist models with the ability to selectively pass information across sequence steps, while processing sequential data one element at a time. Thus, they can model input and/or output consisting of sequences of elements that are not independent. This benefit of recurrent neural network is useful because context plays an important role in handwriting recognition. Unfortunately, the range of contextual information that standard RNNs can access is quite limited. The problem is that the influence of a given input on the hidden layer, and therefore on the network output, either decays or blows up exponentially as it cycles around the network's recurrent connections and is repeatedly scaled by the connection weights. In practice this shortcoming (referred to in the literature as the vanishing gradient problem) makes it hard for an RNN to bridge gaps of more than about 10-time steps

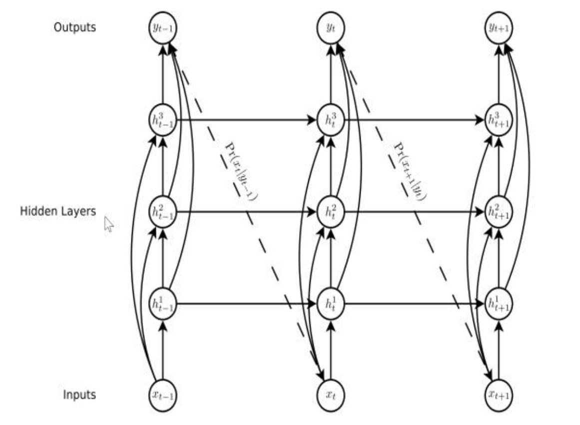

Figure 3. Deep recurrent neural network prediction architecture. [12]

between relevant input and target events [11].

To overcome this vanishing gradient problem, Long Short Term Memory (LSTM) architecture has been designed which consists of multiple recurrently connected subnets, known as memory blocks. Each block contains a set of internal units, or cells, whose activation is controlled by three multiplicative units: the input gate, forget gate and output gate. Figure 3 provides a detailed illustration of an LSTM memory block with a single cell. The gates facilitates storage and access of information over a long periods of time through the cells. For example, the activation of the cell can only be overwritten by new inputs arriving at in the network as long as the input gate is open ie., having and activation close to 1. The activity of the recurrent connection is moderated by the forget gate. The hidden state h(u, v) for position (u, v) of an MDRNN layer is computed based on the previous hidden states h(u−1, v) and h(u, v−1) of both axes and the current input x(u, v) by

$$h(u, v) = \sigma(Wx(u, v)+Uh(u-1, v)+V h(u, v-1)+b),$$

where W, U and V are weight matrices, b a bias vector and σ a nonlinear activation function. Like in the 1D case, MDLSTM introduces an internal cell state for each spatial

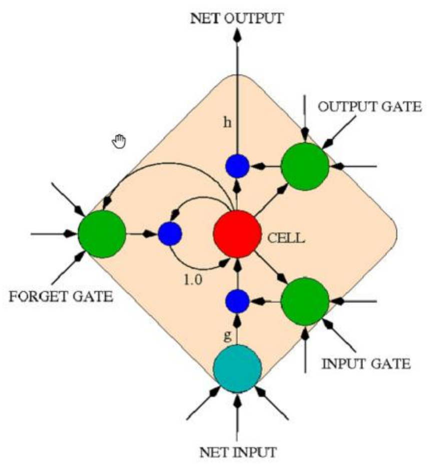

Figure 4. LSTM memory block with one cell [11]

position which is protected by several gates.

Below we discuss some of the recent recurrent neural network-based recognition systems which are the current state of the art in the field.

In Ref [13], Alex Graves and J¨urgen Schmidhuber, proposed an HWR system consisting of Multidimensional Recurrent Neural Network layers that uses Long Short-Term Memory Architecture (LSTM) architecture and Connectionist Temporal Classification (CTC) output layer. This novel architecture outperformed a state-of-the-art HMM-based system on ICDAR 2007 Arabic handwriting recognition competition with 91.4% accuracy compared to 87.2% which was the competition winner. Their proposed system is very general and has been successfully applied to Arabic as well as English. They also claim that the system can be used for almost any supervised sequence labelling task. This was the very first successful RNN model for handwriting recognition on which all other recent RNN systems are based on.

In Ref [14], B. L. D. Bezerra et al., evaluated the performance of Multidimensional Recurrent Neural Network on a publicly available benchmark database called

C-Cube. They also proposed the use of Support Vector Machines in combination with MDRNN to boost its performance. Their experiment was met with encouraging results as they outperformed the MDRNN model and also their proposed model achieved the best rates in the upper and joint databases and is statistically equivalent to the Camastra[15] method in case of the lower letters database. Their experiment gave the best results on C-Cube database Joint Case. The training step of their model is much easier than other methods and classification steps performs faster, due to the fact that MDRNN itself designs and learns everything that is needed from the pixels of the image to distinguish the main differences of the classes.

In Ref [15], Th´eodore Bluche et al., compared Bidirectional LSTM-RNNs with DeepMLPs for unconstrained large vocabulary handwriting recognition on two public databases of multi-line handwritten documents: Rimes and IAM. In this paper they have shown that state-of-art Word Error Rate (WER) can be achieved by both DeepMLPs and RNNs. They also claim that for neural networks, pixel feature values can easily replace handcrafted features.

L. Hidden Markov Models

Hidden Markov Models were first introduced to solve the problem of speech recognition which proved to be highly successful. Since then they have been applied to a number of pattern recognition problems involving processing of sequential data. Considerable progress has been made since their inception in the field of handwriting recognition and a number of systems have been developed based on HMM techniques which have been proven quite successful. The fundamental advantage of Markov-model-based recognizers is that they do not require an explicit segmentation of the data prior to its classification. The recognition is thus performed in a segmentation-free manner, which means that segmentation and classification are integrated.[17]

Hidden Markov Models can be expressed by two-stage stochastic process with hidden states and observable outputs. The first state produces a series of random variables that take on values from a discrete set of states, representing a discrete stochastic process. This is a stationary process in the sense, its statistical properties do not change over time. The last two properties taken together restrict the dependency of the probability distributions of states generated by the random variables to be dependent on the immediate predecessor state only this makes the Markov process to be of first order.[17]

$P(s_t | s_1, s_2, \ldots, s_{t-1}) = P(s_t | s_{t-1})$

This is a probabilistic stage which represents finite automaton. In the second stage, at every time t, and output O_t which only depends on the current state s_t.

$P(O_t | O_1 \ldots O_{t-1}, s_1 \ldots s_t) = P(O_t | s_t)$

To summarize, a first-order hidden Markov model λ is formally defined as consisting of:

– a finite set of states $\{s | 1 \le s \le N\}$,

– a matrix of state transition probabilities4 $A = \{a_{ij} | a_{ij} = P(s_t = j | s_{t-1} = i)\}$,

– a vector of start probabilities $\pi = \{\pi_i | \pi_i = P(s_1 = i)\}$,

and

– state-specific output probability distributions

$\{b_j(O_t) | b_j(O_t) = p(O_t | s_t = j)\}$ for discrete emissions

or $\{b_j(x) | b_j(x) = p(x | s_t = j)\}$ for continuous modelling,

respectively (see below).

Observation sequences can be modelled by two approaches for a practical pattern recognition problem which are model discriminant HMM, and path discriminant HMM. The first one involves building of one or more HMM for each class of pattern. For each observable pattern, calculate the matching score against each model, and classify it to the class whose model leads to the maximum

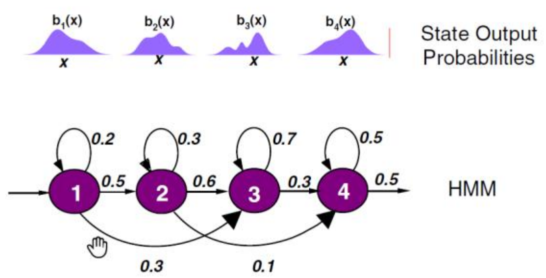

Figure 4. An example of a 4-state, left-to-right, hidden Markov model (HMM) of the type used in the Byblos OCR system. [19]

score. This kind of models have shown successful results in speech recognition. But this type of model is restrictive to smaller vocabularies of up to several hundred words due to prohibition of memory and computation when size scales up to 1000 words or larger vocabularies. The second approach provides a better alternative for larger vocabularies by building one model for all the classes and using different paths, i.e., state sequences to distinguish one pattern from the others.

Below we discuss some of the HMM based recognition systems:

In Ref [20], Moftah Elzobi et al., proposed an HMM based approach that uses an explicit segmentation module in contrast to usual segmentation free approach in mainstream HMM based systems. Their system uses shape representative based feature extractor rather than sliding window-based features. In their system they have constructed an HMM-based threshold model by ergodically connecting all letter models, in order to detect false segmentation as well as non-letter segments. They carried out their experiment using IESK-arDB database for training and testing and IFN/ENIT database for evaluation and achieved best recognition rates of 95% and lowest recognition rates of 86%. In general, they have achieved average recognition rate of 89.6% which is apparently satisfactory compared to existing literature.

In Ref[17], Thomas Plötz and Gernot A. Fink provided a comprehensive survey of the application of Markov models in the research field of offline handwriting recognition, covering both the widely used hidden Markov models and the less complex Markov-chain or n-gram models. They started with introduction of typical architecture of a Markov-model based offline handwriting recognition system and then provided a thorough review of the solutions proposed in the literature for the open problems how to apply Markov-model-based approaches to automatic offline handwriting recognition. Some highlighting points in their discussion are:

- The strength of HMM lies in analysis of sequential data wherein the input length varies greatly
- There are much better approaches for classification of isolated character recognition than HMM
- Majority of HMM based approach perform segmentation free recognition and is considered the most successful approach
- Researchers are focused on estimation of lot of parameters with simple structure instead of complex model architectures which allows for very robust recognition of unconstrained handwriting.
- Hybrid models yield encouraging results

They have identified certain technological trends when analysing recent publications related to the field of Markov-model-based offline HWR which are: Segmentation-free recognition, Simple structure – lots of parameters, Integration of language models, Use of classifier ensembles, Camera-based HWR, Multi-linguality script, independency, Universal toolkits. In the end, they also recommendations to other researchers for reporting results such as: Use well-defined benchmarks, give all necessary technical details Use and use hard task.

In Ref [21], Matthias Zimmermann and Horst Bunke present three different length modelling schemes to optimize the number of states in left-to-right HMMs. In their first method, they propose the fixed length modelling scheme where each character model is assigned the same number of states. Their second method, considered is the Bakis length modelling where the number of model states is set to a given fraction of the average number of observations of the corresponding character, while in the third one, the number of model states is set to a specified quantile of the corresponding character length histogram which is called quantile length modelling. They have also drawn comparison of the different length

modelling schemes which has been carried out with a handwriting recognition system using off-line images of cursively handwritten English words from the IAM database achieving a recognition rate of 61% for the fixed length modelling. They also claim that using Bakis or quantile length modelling the word recognition rates could be improved to over 69%.

5. CONCLUSION

In this paper three of the most prominent approaches for the problem of Offline Handwriting Recognition briefly summarized. We conclude that out of all the methods available, systems based on different approaches of recurrent neural networks involving Long Short Term Memory architecture seem most promising due to being able to retain long term dependencies which can be exploited to extract contextual information that can help in more robust predictions. Convolutional Neural Networks have been proven to be superior compared to traditional feature extraction techniques such as handcrafted features. And while HMM based have been quite successful, their inability to make use of previous context makes them relatively inadequate for very large vocabularies or sentence level recognition.

REFERENCES

[1] T. Caesar, J. M. Gloger and E. Mandler, "Preprocessing and feature extraction for a handwriting recognition system," Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR '93), Tsukuba Science City, Japan, 1993, pp. 408-411.

[2] R.M. Bozinovic, S.N. Srihari, Off-line cursive script word recognition, IEEE Trans. Pattern Anal. Mach. Intell. 11 (1) (1989) 69–83.

[3] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," in Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.

[4] Phạm D.V. (2012) Online Handwriting Recognition Using Multi Convolution Neural Networks. In: Bui L.T., Ong Y.S., Hoai N.X., Ishibuchi H., Suganthan P.N. (eds) Simulated Evolution and Learning. SEAL 2012. Lecture

Notes in Computer Science, vol 7673. Springer, Berlin, Heidelberg

[5] Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review by Waseem Rawat and Zenghui Wang in Neural Computation 2017 29:9, 2352-2449

[6] T. Bluche, H. Ney and C. Kermorvant, "Feature Extraction with Convolutional Neural Networks for Handwritten Word Recognition," 2013 12th International Conference on Document Analysis and Recognition, Washington, DC, 2013, pp. 285-289.

[7] D. Suryani, P. Doetsch and H. Ney, "On the Benefits of Convolutional Neural Network Combinations in Offline Handwriting Recognition," 2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), Shenzhen, 2016, pp. 193-198.

[8] P. Y. Simard, D. Steinkraus and J. C. Platt, "Best practices for convolutional neural networks applied to visual document analysis," Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings., Edinburgh, UK, 2003, pp. 958-963.

[9] Hu, Jie; Shen, Li; Albanie, Samuel; Sun, Gang; Wu, Enhua (2019). "Squeeze-and-Excitation Networks". IEEE Transactions on Pattern Analysis and Machine Intelligence

[10] Botvinick, M. M., & Plaut, D. C. (2006). Short-term memory for serial order: A recurrent neural network model. Psychological Review, 113(2), 201–233. doi:10.1037/0033-295x.113.2.201

[11] Liwicki M., Graves A., Bunke H. (2012) Neural Networks for Handwriting Recognition. In: Ogiela M., Jain L. (eds) Computational Intelligence Paradigms in Advanced Pattern Classification. Studies in Computational Intelligence, vol 386. Springer, Berlin, Heidelberg

[12] Alex Graves. Generating Sequences With Recurrent Neural Networks. arXiv:1308.0850

[13] Graves, A., Schmidhuber, J.: Offline Handwriting Recognition with Multidimensional Recurrent Neural Networks. In: Adv. in Neural Information Proc. Syst., pp. 545–552 (2009)

[14] B. L. D. Bezerra, C. Zanchettin and V. Braga de Andrade, "A Hybrid RNN Model for Cursive Offline Handwriting Recognition," 2012 Brazilian Symposium on Neural Networks, Curitiba, 2012, pp. 113-118.

[15] Camastra, F.: A SVM-Based Cursive Character Recognizer. Pattern Recognition 40(12),3721–3727 (2007)

[16] Bluche T., Ney H., Kermorvant C. (2014) A Comparison of Sequence-Trained Deep Neural Networks and Recurrent Neural Networks Optical Modeling for Handwriting Recognition. In: Besacier L., Dediu AH., Martín-Vide C. (eds) Statistical Language and Speech Processing. SLSP 2014. Lecture Notes in Computer Science, vol 8791. Springer, Cham

[17] Plötz, T., & Fink, G. A. (2009). Markov models for offline handwriting recognition: a survey. International Journal on Document Analysis and Recognition (IJDAR), 12(4), 269–298. doi:10.1007/s10032-009-0098-4

[18] Mou-Yen Chen, Amlan Kundu and Jian Zhou, "Off-line handwritten word recognition using a hidden Markov model type stochastic network," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, no. 5, pp. 481-496, May 1994.

[19] Natarajan, P., Saleem, S., Prasad, R., MacRostie, E., & Subramanian, K. (n.d.). Multi-lingual Offline Handwriting Recognition Using Hidden Markov Models: A Script-Independent Approach. Lecture Notes in Computer Science, 231–250. doi:10.1007/978-3-540-78199-8_14

[20] M. Elzobi, A. Al-Hamadi, L. Dings, M. Elmezain and A. Saeed, "A Hidden Markov Model-Based Approach with an Adaptive Threshold Model for Off-Line Arabic Handwriting Recognition," 2013 12th International Conference on Document Analysis and Recognition, Washington, DC, 2013, pp. 945-949.

[21] M. Zimmermann and H. Bunke, "Hidden Markov model length optimization for handwriting recognition systems," Proceedings Eighth International Workshop on Frontiers in Handwriting Recognition, Niagara on the Lake, Ontario, Canada, 2002, pp. 369-374.

[22] Alessandro Vinciarelli. A survey on off-line Cursive Word Recognition. Elsevier, Pattern Recognition, Volume 35, Issue 7, July 2002, Pages 1433-1446