January-February 2020
ISSN: 0193-4120Page No. 6773 - 6777

Hand Written Characters Recognition Using Deep Learning

[1]G. Pavan, [2]J. Rene Beulah, [3]M. Nalini

[1]UG Scholar, [2,3]Assistant Professor
Department of Computer Science and Engineering, Saveetha School of Engineering,
Saveetha Institute of Medical and Technical Sciences, Chennai, India
[1]pavanpollard333@gmail.com, [2]renebeulah@gmail.com, [3]nalini.tptwin@gmail.com

Article Info
Volume 82
Page Number: 6773 - 6777
Publication Issue:
January-February 2020

Article History
Article Received*: 18 May 2019*
Revised*: 14 July 2019*
Accepted*: 22 December 2019*
Publication*: 01 February 2020*

Abstract
This paper provides a new solution to conventional techniques for handwriting recognition using Deep learning ideas and computer vision. An expansion of MNIST digits dataset was used, called the Emnist dataset. This contains 62 classes in both upper and lower case with 0-9 digits and A-Z characters. An application has been developed for Android to identify handwritten text and transform it to digital form using Convolutional Neural Networks, abbreviated as CNN, to identify and detect text. We pre-processed the dataset before that, and used different filters to it. Using Android Studio we created an android application and connected our text recognition program to handwriting utilizing tensorflow libraries. The application's layout was kept simple for display purposes. The experienced keras graph is used through a protobuf file and tensorflow interface to anticipate alphanumeric characters drawn with a finger.

Keywords: *Deep Learning, Hand Written characters, Training.*

1. Introduction

In spite of the abundance of technical writing tools, several people even now traditionally select to use pen and paper to take their notes. The text does, however, have shortcomings. It is difficult to efficiently store and access physical documents, search through them efficiently and share them with others. Therefore, due to the fact that records are never found Transferred to digital format, a lot of important information is lost or not review. We have decided to deal with There is a problem in our project because we consider it much easier to manage digital text Written lessons will help people to use them more efficiently, Searching, sharing and analyzing their documents, while still allowing them to use their favorite writing method [1]. The purpose of this project is to explore additional work Classify handwritten text and in digital format convert handwritten text. Handwritten text is a really common term, and for our aim we needed to narrow its scope Project, defining the meaning of handwritten text. In such a project we have challenged to categorize the picture of any handwritten word that may be in the form of cursive or block writing. This scheme can be mixed with algorithms fragmenting

word images in a line drawing, which in move can be mixed into a whole handwritten page image with algorithms fragmenting line images. Our project takes the form of end-user delivery for these added layers and will be a complete functional model that will help solve the user's issue of converting Prompting handwritten records into digital format [2]. To take a photo of the user notes page. Pay attention to that too while we need to have some extra layers on head of the Model to make a complete functional distribution for end users, we believe that the categorization part is the most interesting and challenging part of the issue and that is why we chose to deal with it rather than dividing the lines [3]. In words, lines of records etc. We face this problem with full pictures of the word. Since CNNs work much better with raw input pixels from picture characteristics or sections [4].Looking at our results using entire word images, we attempted creation by removing the characters from each word image and then categorizing them. For every character to independently recreate an whole word. In general, our model takes output a picture of a word and the name of a word in both of our strategies.

2. Proposed Method

2.1 Architecture

We have used EMNIST's dataset; there are many achievements that have been attained using this dataset. When using Deep Learning, handwritten text recognition is created however, their accuracy was actually low or they had a relatively tiny dataset by the line Iqvil. In this paper, the use of OCR is discussed such as speech

January-February 2020
ISSN: 0193-4120Page No. 6773 - 6777

recognition, radio frequency, Vision systems, magnetic stripes, bar codes and optical mark reading. A famous machine learning task MNIST is categorizing datasets, which are datasets of numbers. Best Practices for Constitutional Nerve the Network Applied for Visual Document Analysis by Simar, Stancras and Plotte is a beneficial paper. Understanding the use of complex neural networks (CNNs). For word recognition, a paper by Pham et Al., Used a 2-layer CNN, fed with a short-large- in a bidirectional recurrent neural network (RNN). Term memory (LSTM) cells. According to us, the best model implemented is that by Graves and Schmiduber with a multidimensional RNN structure. M. J. Another paper on 'Handwritten Text Recognition' by. Castro-Bleda dealt with the dataset with slotted words and corrected them during pre-processing. [5] Development of English handwritten recognition by Teddy Surya using Deep Neural Network and Ahmed Fakhrur uses a deep neural network model that consists of two encoding layers and a softmax layer EMNIST dataset. Their precision using DNN was better than the previously suggested Pattern Net and feed forward net ANN (Artificial Neural Network). Handwritten text recognition can also be obtained Relaxation based on Conversational Neural Networks (R-CNN) and optionally trained waivers Steadfast neural network (ATRCNN) performed by Chunpeng Wu and Wei Fan [6]. Our model attained More than 87 percent accuracy utilizing conventional neural networks from the Kerus library stored and submitted as a black and white document (grayscale or bitmap.) If a graphic appears in colour, it should be submitted as a color document for RGB [7].

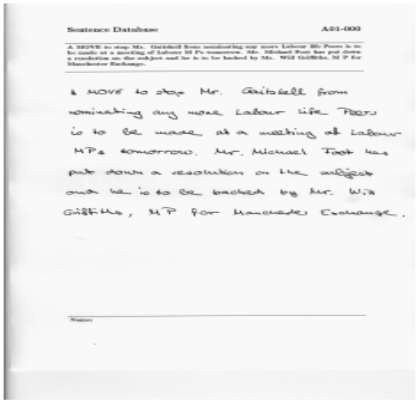

Figure 1: An example form from the IAM Handwriting dataset. Word images in the dataset were extracted from such forms.

2.2 Methodology

Our project's aim is to create an interface that can be used to recognize handwritten characters from users. To get a better precision, we approached our problem by using Convolutional Neural Networks. Many works to improve the accuracy of alphanumeric character prediction has been undertaken. That will be part of our work to some extent. Yet our primary focus will be to provide a GUI that can be used to predict characters for further use with ease. We schedule to use tensorflow [8] and keras [9] to do so. First, we will describe a model to be trained with the Emnist dataset that includes more than 690,000 train pictures and will be evaluated again using Emnist's test data set. We will then freeze the keras graph and connect it to our Android application with an easy layout that gets a hand-drawn user input and predicts the alphanumeric character. We'll be working on python later to segment characters from a word picture and anticipate every character using our model. The IAM handwriting dataset [10] was our main resource to train our handwriting identifiers. This dataset contains more than 1500 handwritten text, where a type there is a paper with text lines from more than 600 writers, with 5500 + sentences and 11500 + words. Word Was then manually fragmented and verified; all connected form label metadata are given in the corresponding XML files. The source text was depend on the corpus of Lancaster-Oslo / Bergen (LOB), which contains texts of complete English sentences with more than 1 million words in total. The database It also contains 1,066 forms produced by about 400 Various authors. This database has shown its width, depth, and quality serves as the basis for many handwriting recognition tasks and motivates us for those reasons Option of IAM handwriting dataset as our source Training, validation, and test data for our model [11]. Last but At least, large datasets with deep learning — not even with many Pre-trained models - are very important and there are examples of 100K + words in this dataset that meet those requirements (Deep learning model requires at least 105 - 106 Despite being in a position to perform well in training instances, transfer learning).

2.3 Rotating Image

Even though the pictures for each word in our dataset are different, these pictures had some words Slightly bent. The reason for this was the contributors. The dataset was told to write on empty paper in a more bent fashion, without any lines and a few words being written. This opportunity occurs very often in everyday life or the page has no lines, so we chose to make the issue more robust by rotating a picture to the right with a very small angle with random probability to our training data and the addition of that image in our training set. This information enhancement method assisted us to create our model to a certain small but consistent details more robust. Come on with our test set [12].

2.4 Zero-Centering Data

$$X^{(k)}_{centered} = X^{(k)} - \frac{1}{N}\sum_{k=1}^{N}\frac{1}{H*W}\sum_{i=1}^{H}\sum_{j=1}^{W}X^{(k)}_{i,j}$$

We center our information before practice by deducting the mean data set pixel values from the picture of pixel values. This is required because we are using a single learning process. The rate we want to be on a scale relative to the weight between each other when we

change our weight, so that all weights are impacted by this multiplication in the same relative way when we multiply with a single learning rate. If we have not done anything, some weight on each update will be greatly amplified and others will be rare [13].

3. Methods

3.1 Vocabulary Size

As can be seen in our paper's data section, we contained a huge number of individual terms in our dataset. However, few word pictures appeared only in our dataset. Time which made training us on such pictures very difficult. This problem, together with the reality that our dataset already had a huge glossary, motivated us to remove a few Words in our dataset and not include the words our training / verification / testing dataset that we were going to use with our design. So we restricted our data with words Pictures appearing in our dataset (previously) split into train and validation sets at least twenty times. Nonetheless, we had 4000 individual words with at least 20 occurrences in our dataset [14].To speed up our preparation, we chose to reduce 50 words to the size of our terminology, which still takes 5-6 hours to learn and verify. Do not depend on our model and algorithms Hardcode number of pictures and must be operated with any instances but we chose to narrow it down Number of words for effectiveness needs.

3.2 Word-Level Classification

First we create a vocabulary based on a selection of 50 words in the word-level categorization model. Our dataset contains at least 20 occurrences. We practiced with several CNN frameworks in the Word classification: VGG-19, RESNET-18, and RESNET-34. Architecture of the VGG Steadfast Network was one of the first very deep neural traps to attain cutting-edge results on main deep learning tasks. Defining standard practice on time, V.G.G. Very small comfortable filter (3 x 3) and a less number of receptive field channels in return [15]. Their network is increasing in depth for computational balance Cost for the ImageNet 2014 Challenge. For various iterations of his model, walking by conventional 3-7 layers of prior CNN up to 16-19 layers, not just his model Ranked first and second in localization. Residual Networks (RESNETs) were found to top VGG First place in the ImageNet challenge [17], respectively, but also to generalize well to other computer vision tasks. Identifying that it was difficult to train very deep learning networks in part Due to the stagnation of gradient flow in the first layers Web. That at Al created the concept of residual Layer, which in turn learns different functions with respect to the input of the function's layers. Hence, a Very deep residual network will have a layer option a zero residual learning (this tendency can be applied to regular regularization) and thus conservation Input and preservation of gradients for the first layers during back propagation [18].This design permitted RESNET 100 + layers to train many more layers in terms of capability and prior intensive learning models and simultaneous Parameterization. RESNET dates from this paper Intensive education among the most likely candidates attempted models for computer vision projects.

3.3 Training

After softmax activation we practiced all of our models with the ADAM adapter on the cross-entropy loss function. Otherwise, we preserved the same architecture for the VGG and RESNET models except for the last conversion Fully connected output layer rather than mapping classes to number (Word / Letter vocabulary). We trained the word and character classification from scratch. For the character segmentation model, we used a fine Tus act model [19].

Figure 2: These are the filters from the first convolutional layer of the RESNET-18 word segmentation model with the example word ”much”

Softmax Loss:

$$L_i = -log(\frac{e^{f_{W_i}}}{\sum_j e^{f_j}})$$

3.4 Segmentation

For word-level categorization, we suspected that our performance suffered due to the huge softmax layer output size (our training was more than 10,000 words in vocabulary and English language alone) and complexity in properly identifying pictures of words. We chose that the character level categorization may be a more realistic method because has a fairly broad character vocabulary comparatively much shorter than the same broad word vocabulary (the letters A-Za-z0-9 are only 62 different symbols), greatly limiting computational difficulty. Of softmax also, recognition of a character a picture is an easier task than identifying a word an image due to the limited range of characters. However, the first major challenge we will face. The order to test this approach would be division of the word Pictures in their constituent character images. The second main challenge will be to identify the word break. The image and string form the words by mixing the letters that are continuously identified between these words. We will do Already address these challenges in this section. In To execute this function, we used the new CNN / LSTM engine based on the neural network Tesser act 4.0 [20]. This model is configured as a textline identifier originally developed by HP and is now maintained by Google which can recognize more than 100 languages out of the box. Tesser act model for Latin language, including English about 400000 textlines were trained 4500 fonts. Then we

January-February 2020
ISSN: 0193-4120Page No. 6773 - 6777

removed the parameters on our IAM handwriting dataset for this pretested model. After U.S Financed the model, we divided the original word input images into their envisaged component character images, feeding into images of these output characters In our character-level classification.

3.5 Character Level Classification

The character-level classification model was similar to the word-level classification model. The main difference included passing in character pictures instead of words.

Images, using a character level vocabulary instead of one Training of word-level vocabulary, and a separate paratrization of each version of our very deep learning model. The architectures of these models were otherwise

Similar to our word-level model.

Architecture	Training Accuracy	Validation Accuracy	Test Accuracy
VGG-19	28%	22%	20%
RESNET-18	31%	23%	22%
RESNET-34	35%	27%	25%
Char-Level	38%	33%	31%

Figure 3: Word Level and Character Level Classification Results

4. Conclusion

Using modern technology such as neural networks to apply intensive learning to solve common tasks which is done by any human with a blink of the eye as if the recognition of the text just scratches the surface. The ability behind machine learning. This technique has endless possibilities and applications. Worked similar to traditional OCR biometric devices. Image sensor technology has been used to gather match points of physical attributes and then transform it to a database of known kinds. However, with the help of modern technologies such as conventional neural networks, we can scan and understand Words with precision that has never been seen before in history. We are using the EMNIST data set to train our model and finally tested various optimizers to choose adamax as it does not only reach high precision on our test data with each Era, but also on our train data. Accurate text is another application of OCRs to help Partially spotted and blinded in Braille's absence. By also merging a simple text for speech into the application the user of the module can point their phone to any text that will then interpret the user's text. A dedicated tool with more sophisticated image recognition can also be designed for that purpose. System that can identify objects and tell the user in which direction and how many steps to take when to stop and turn. The EMNIST dataset, a suite of six datasets, has grown significantly Employing only MNIST datasets poses a challenge. Even though the framework of the EMNIST dataset is Similar to MNIST, it provides image samples and output classes and an even higher number More complex and varied classification tasks. So the use of this as the backbone of our project was clear. It would be practically impossible to achieve this accuracy without the use of an EMNIST data set. Our new Android app needs the user to drag and write text to the screen and then examine it to recognize the character of the alphanumeric. The app can be further developed to import images Identify the gallery and the text in those images in the user's device. May be another development Converting text to speech to further enhance mobile app applications. Can be android app Goggles was further developed using the Cloud Natural Language API which provides natural language Understanding technologies, sentiment analysis, entity recognition, entity sentiment analysis, and Interpret text to further understand it by giving dictionaries that improve text annotations errors made by the model to supply a meaningful result. Google's Cloud Vision API may also be used to improve data precision and even identify various objects.

References

[1] Alex Graves, Santiago Fernndez, Faustino Gomez, Jrgen Schmidhuber, Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks, Proceedings of the 23rd international conference on Machine learning. 2006

[2] Convolutional Neural Network Benchmarks: https://github.com/jcjohnson/cnn-benchmarks.

[3] Elie Krevat, Elliot Cuzzillo. Improving Off-line Handwritten Character Recognition with Hidden Markov Models.

[4] Fabian Tschopp. Efficient Convolutional Neural Networks for Pixelwise Classification on Heterogeneous Hardware Systems

[5] Mail encoding and processing system patent: https://www.google.com/patents/US5420403

[6] H. Bunke1, M. Roth1, E.G. Schukat-Talamazzini. Offline Cursive Handwriting Recognition using Hidden Markov Models.

[7] K. Simonyan, A. Zisserman Very Deep Convolutional Networks for Large-Scale Image Recognition arXiv technical report, 2014

[8] Lisa Yan. Recognizing Handwritten Characters. CS 231N Final Research Paper.

[9] Oivind Due Trier, Anil K. Jain, Torfinn Taxt. Feature Extraction Methods for Character Recognition–A Survey. Pattern Recognition. 1996

[10] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. Neural Information Processing Systems (NIPS), 2015

[11] How many words are there in the English language?:

January-February 2020
ISSN: 0193-4120Page No. 6773 - 6777

https://en.oxforddictionaries.com/explore/howmany-words-are-there-in-the-english-language

[12] Thodore Bluche, Jrme Louradour, Ronaldo Messina. Scan, Attend and Read: End-to-End Handwritten Paragraph Recognition with MDLSTM Attention.

[13] T. Wang, D. Wu, A. Coates, A. Ng. "End-to-End Text Recognition with Convolutional Neural Networks" ICPR 2012.

[14] 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.

[15] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner: Gradient-Based Learning Applied to Document Recognition, Intelligent Signal Processing, 306-351, IEEE Press, 2001

[16] Nalini, M. and Anbu, S., "Anomaly Detection Via Eliminating Data Redundancy and Rectifying Data Error in Uncertain Data Streams", Published in International Journal of Applied Engineering Research (IJAER), Vol. 9, no. 24, 2014.

[17] Shiny Irene D., G. Vamsi Krishna and Nalini, M., "Era of quantum computing - An intelligent and evaluation based on quantum computers", Published in International Journal of Recent Technology and Engineering (IJRTE), Vol. 8, Issue no.3S, pp. 615- 619, October 2019.[DOI> 10.35940/ijrte.C1123.1083S19]

[18] V. Padmanaban and Nalini, M. , Adaptive Fuel Optimal and Strategy for vehicle Design and Monitoring Pilot Performance, Proceedings of the 2019 international IEEE Conference on Innovations in Information and Communication Technology,Apr 2019. [DOI>10.1109/ICIICT1.2019.8741361]

[19] J. Rene Beulah and D. Shalini Punithavathani (2017). "A Hybrid Feature Selection Method for Improved Detection of Wired/Wireless Network Intrusions", Wireless Personal Communications, vol. 98, no. 2, pp. 1853-1869 (Springer).

[20] J. Rene Beulah and Dr. D. Shalini Punithavathani (2015). "Outlier Detection Methods for Identifying Network Intrusions – A Survey", *International Journal of Applied Engineering Research*, Vol. 10, No. 19, pp. 40488-40496.

[21] J. Rene Beulah, N. Vadivelan and M. Nalini (2019). "Automated Detection of Cancer by Analysis of White Blood Cells", International Journal of Advanced Science and Technology, vol. 28, No. 11, pp. 344-350.