International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 12 | December - 2022 Impact Factor: 7.185 ISSN: 2582-3930

HANDWRITTEN TEXT RECOGNITION USING MACHINE LEARNING

R. MADHAVA REDDY, K. DINESH REDDY, CH. HARI BABU

Mr. P. DURGA PRASAD (Guide)

Sreenidhi Institute of Science and Technology

Hyderabad

ABSTRACT:

In this rapidly evolving world everything is getting digitalized, Exchange of information plays a crucial role in any work. Most of the times we may not find the facility to create a document so we use pen and paper but people may not understand everything expressed on paper due to many reasons line bad penmanship etc. So we are building a platform through which one could convert the handwritten text to editable text. Our main aim is to classify the words directly and character segmentation. We use CNN for this process. We use long term memory networks (LSTM) with convolution to construct bounding boxes for each character. We divide the each given character and classify them and reconstruct for each word of neural networks for Handwriting character recognition.

INTRODUCTION

Despite the abundance of technological writing tools, many people still choose to take their notes traditionally: with pen and paper. However, there are drawbacks to handwriting text. It's difficult to store and access physical documents in an efficient manner, search through them efficiently and to share them with others. Thus, a lot of important knowledge gets lost or does not get reviewed because of the fact that documents never get transferred to digital format. We have thus decided to tackle this problem in our project because we believe the significantly greater ease of management of digital text compared to written text will help people more effectively access, search, share, and analyze their records, while still allowing them to use their preferred writing method.

The aim of this project is to further explore the task of classifying handwritten text and to convert handwritten text into the digital format. Handwritten text is a very general term, and we wanted to narrow down the scope of the project by specifying the meaning of handwritten text for our purposes. In this project, we took on the challenge of classifying the image of any handwritten word, which might be of the form of cursive or block writing. This project can be combined with algorithms that segment the word images in a given line image, which can in turn be combined with algorithms that segment the line images in a given image of a whole handwritten page. With these added layers, our project can take the form of a deliverable that would be used by an end user, and would be a fully functional model that would help the user solve the problem of converting handwritten documents into digital format, by prompting the user to take a picture of a page of notes. Note that even though there needs to be some added layers on top of our model to create a fully functional deliverable for an end user, we believe that the most interesting and challenging part of this problem is the classification part, which is why we decided to tackle that instead of segmentation of lines into words, documents into lines, etc. We approach this problem with complete word images because CNNs tend to work better on raw input pixels rather than features or parts of an image. Given our findings using entire word images, we sought improvement by extracting characters from each word image and then classifying each character independently to reconstruct a whole word.

EXISTING SYSTEM

The first prominent piece of OCR software was invented in 1974 as the software allowed for recognition for any font Next approach uses letters as a state which then allows for the context of the character to be accounted for when determining the next hidden variable. This led to higher accuracy compared to both feature extraction techniques and the Naive Bayes approach. The main drawback was still the manual extraction features, which requires prior knowledge of the language and was not particularly robust to the diversity and complexity of handwriting.

PROPOSED SYSTEM

We neural networks that can be used to combine the tasks of localizing text in an image along with understanding what the text is We use CNN, RNN, CTC neural networks in our project. Using deep convolutional neural architectures and attention mechanisms and recurrent networks show good results. We use word images because CNNs tend to work better on pixels rather than parts of an image. We use RNN recurrent neural networks that stores sequential data for recognition. We perform connectionist temporal classification to train RNN models. Results have shown that Long Short-Term Memory (LSTM) based OCR yields low error rates which we use in our project.

WHAT IS CNN?

Convolutional Neural Networks, often known as CNNs, is a type of neural network designed for deep learning algorithms that are particularly useful for tasks involving pixel data and image recognition.

CNN is the most effective neural network for detecting and recognizing the item from the input. We employed it in our project due to its correctness.

Inside Of CNN:

The input must pass through each of CNN's three layers in order to produce its output. The CNN's three tiers are

- Convolutional layer
- Pooling Layer
- Fully Connected Layer

Convolutional Layer: In this layer, most computing takes place. A kernel or filter inside this layer moves over the image's receptive fields during the convolution process to determine if a feature is present.

The kernel traverses the entire picture over several numbers of iterations. A dot product between the input pixels and the filter is calculated at the end of each cycle. A feature map or convolved feature is the result of the dots being connected in a certain pattern. In this layer, the picture is ultimately transformed into numerical values that the CNN can understand and extract pertinent patterns from. In this layer, the picture is ultimately transformed into numerical values that the CNN can understand and extract pertinent patterns from

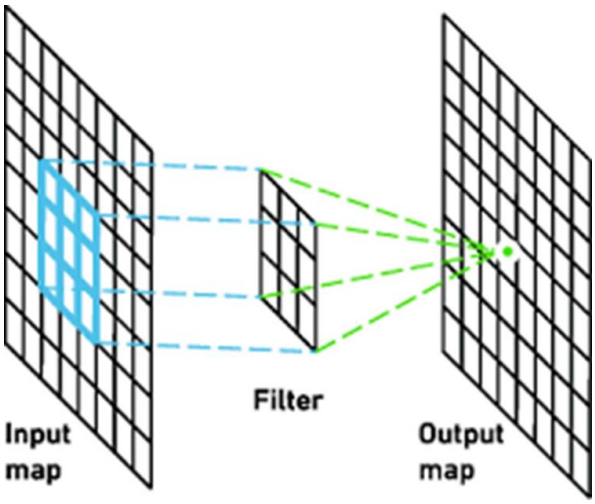

Fig2: Convolutional Layer

Pooling Layer: The pooling layer similar to the convolutional layer sweeps a kernel or filter over the input image. Contrary to the convolutional layer, the pooling layer has fewer input parameters but also causes some information to be lost. Positively, this layer simplifies the CNN and increases its effectiveness.

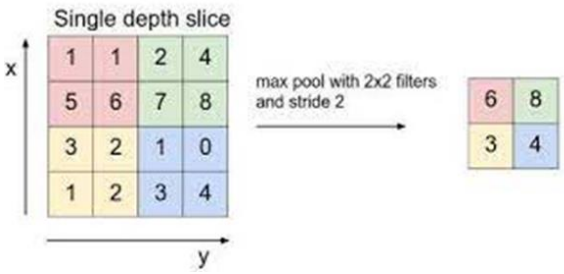

Fig 3: Pooling Layer

Fully Connected Layer: Based on the characteristics gathered in the preceding layers, picture categorization in the CNN takes place in the FC layer. Fully lined in this context indicates that every activation unit or node of the subsequent layer is connected to every input or node from the preceding layer.

WORKING OF CNN

Multiple layers of a CNN are possible, and each layer trains the CNN to recognize the many aspects of an input picture. Each picture is given a filter or kernel to create an output that grows better and more detailed with each layer. The filters may begin as basic characteristics in the bottom levels.

To examine and identify characteristics that specifically reflect the input item, the complexity of the filters increases with each additional layer. As a result, after each layer, the output of each convolved picture becomes the input for the following layer. The CNN recognizes the picture or objects it represents in the final layer, which is an FC layer.

The input image is processed via a number of different filters during convolution. Each filter performs its function by turning on specific aspects of the image, after which it sends its output to the filter in the subsequent layer. The procedures are repeated for dozens,

hundreds, or even thousands of layers as each layer learns to recognize various characteristics. Finally, the CNN is able to recognize the full object thanks to the picture data flowing via its numerous layers.

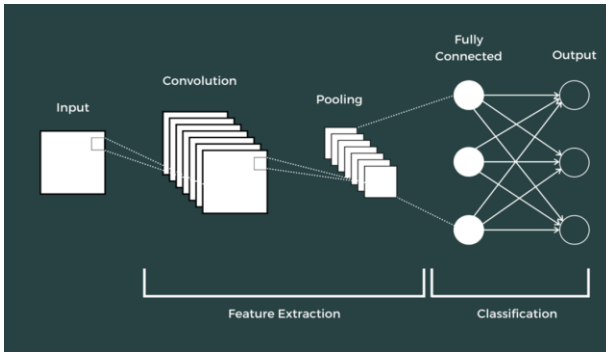

Fig4: CNN layers

ADVANTAGE OF CNN

CNN is particularly renowned for its results' great accuracy despite the volume of data. As CNN advances, it removes all the extraneous data from the input provided in order to improve the accuracy of the outputs. This is because CNN has learned from its previous phase. When compared to single-layered neural networks, many layers significantly boost the output accuracy. Any device can easily implement CNN construct models; mobile phones can even run them. CNN is employed in many different applications, including automotive and face recognition.

SOFTWARE AND HARDWARE REQUIREMENTS

In this project we have used Windows 10, Intel Core i7-9700 CPU of 3.00 GHz, 32.0 GB RAM, and NVIDIA GeForce RTX 2080 graphics card.

CONCLUSION

We discussed a NN which is able to recognize text in images. The NN consists of 5 CNN and 2 RNN layers and outputs a character-probability matrix. This matrix is either used for CTC loss calculation or for CTC decoding. An implementation using TF is provided and some important parts of the code were presented. Finally, hints to improve the recognition accuracy were given.

FUTURE SCOPE

Firstly, to have more compelling and robust training, we could apply additional preprocessing techniques such as jittering. We could also divide each pixel by its corresponding standard deviation to normalize the data. Next, given time and budget constraints, we were limited to 20 training examples for each given word in order to efficiently evaluate and revise our model. Another method of improving our character segmentation model would be to

move beyond a greedy search for the most likely solution. We would approach this by considering a more exhaustive but still efficient decoding algorithm such as beam search. We can use a character/word-based language-based model to add a penalty/benefit score to each of the possible final beam search candidate paths, along with their combined individual softmax probabilities, representing the probability of the sequence of characters/words. If the language model indicates perhaps the most likely candidate word according to the softmax layer and beam search is very unlikely given the context so far as opposed to some other likely candidate words, then our model can correct itself accordingly.

ACKNOWLEDGMENTS

We would like to express our special thanks to our mentor Mr. P. Durga Prasad who gave us a golden opportunity to do this wonderful project on this topic which also helped us in doing a lot of research and we came to know about so many new things. We are really thankful to them.

Secondly, We would also like to thank my friends who helped us a lot in finalizing this project within the limited time frame.