Matthew Goldgirsh

CS 4100 Final Project

Optical Character Retrieval Evaluation

Professor Sloan

Introduction

Optical Character Recognition (OCR) is becoming more predominantly used in processing PDF documents and summarization of PDF-type documents. To process a PDF file as a summary of text OCR is typically used, the idea that characters can be recognized from an image or PDF and then represented as a letter.

I have always been infatuated with the idea of OCR and its potential uses to speed up document processing and hence my interesting in implementing an OCR system and learning the nuances on my own.

Because of my interest and contemporary enthusiasm about OCR this paper will explore various kinds of OCR processing and evaluate the effectiveness of each model both on predetermined datasets of documents and then determine the validity of OCR usage in the future of document processing and automation of mundane processes.

Methodology

This project will explore various OCR models and evaluate their effective based on the proportion of characters they correctly recognize. The first model this project will analyze will be the EasyOCR pretrained OCR model which is based on a combination of convolutional neural networks (CNN) as well as Long Short-Term Memory networks (LSTM's). By utilizing

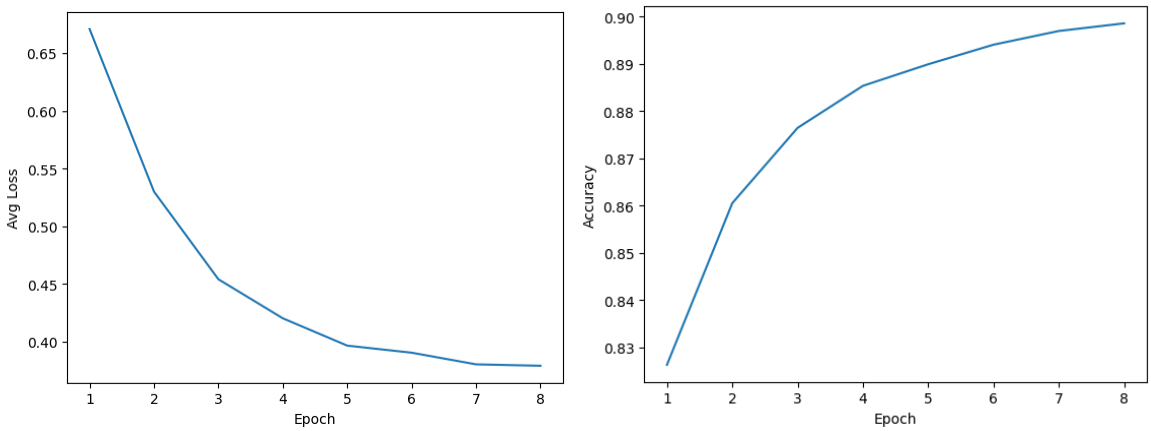
both CNN and LSTM networks, EasyOCR can accomplish very effective character recognition through the convolutional network model, as well as very important context understanding and sequence labeling with the LSTM model. This model was chosen to serve as a benchmark for a well-known OCR model to compare a locally created OCR model against.

Aside from the EasyOCR model, a simpler OCR model was constructed using a standard convolutional neural network and innovative approaches for document segmentation. The convolutional neural network was trained on a character dataset taken from Kaggle over eight epochs to have a solid character recognition model to use in the OCR processing. Furthermore, to fully evaluate the discrepancies between the simple OCR model and EasyOCR, two different image segmentation algorithms were implemented to fully demonstrate the capabilities of OCR with the simple OCR model. One image segmentation algorithm is a brute force approach, taking as many smaller images as possible from the larger document and running OCR individually on each of the images, while the other segmentation model is more elaborate where the segments are made based on the threshold of black pixels in the current segment of the document.

Both these segmentation models will be evaluated with the baseline of EasyOCR to determine how effective they are at OCR processing.

Experimentation

First the simpler OCR model's CNN needed to be trained to recognize characters. This training was completed, and the following graphs were generated.



The left graph shows the avg loss at the end of each epoch steadily decreasing for the simple OCR model. The right graph shows the opposite relation with accuracy as accuracy slowly goes up to around 90% at the end of training, showing that the CNN used for the simple OCR model is highly effective at recognizing characters.

Since the accuracy of the CNN for the simple OCR model is very high when used only on characters, if the simple OCR model has an effective segmentation approach the quality of the OCR will be high as well. Two segmentation approaches were utilized one rudimentary approach and the other a slightly more complex approach that should perform better at recognizing characters. These approaches where then compared with the EasyOCR predefined model to evaluate.

Results and Evaluation

The evaluation was done in a bag-of-words type manner where predictions were split up into unigrams and the frequency of characters was compared with the actual frequency of character on the document. There were 3 documents evaluated which were labeled by hand, 2 taken from the *Artificial Intelligence - A Modern Approach (3rd Edition)*

Textbook and the other taken from a Google Image Search.

In the case of the EasyOCR model the results were as following:

```
Difference between documents/doc3.webp and ocr is: 0.037037037037037035
Difference between documents/doc1.png and ocr is: 0.03724247226624406
Difference between documents/doc2.png and ocr is: 0.007042253521126761
```

These results show that the EasyOCR predetermined model was very effective in finding the text in the documents as the difference between what EasyOCR computed and the actual text was as low as 0.007

However, running the simple OCR model on the same dataset with rudimentary (brute force) segmentation provided more interesting results.

```
Difference between documents/doc3.webp and ocr is: 0.9629629629629629
Difference between documents/doc1.png and ocr is: 0.9080824088748018
Difference between documents/doc2.png and ocr is: 0.9172535211267606
```

These results show that the simple OCR with brute force segmentation was hardly effective at performing character recognition in the large document sense. Furthermore, when comparing with the results from the enhanced segmentation model (shown below) it is evident that the simple OCR model has a very minimal accuracy in processing documents even when the segmentation algorithm was improved.

```
Difference between documents/doc3.webp and ocr is: 0.9351851851851852
Difference between documents/doc1.png and ocr is: 0.8700475435816165
Difference between documents/doc2.png and ocr is: 0.8996478873239436
```

The results show that the lowest difference between the documents and the simple OCR model was around 0.87 meaning that around 87% of the text recognized by the simple OCR model was missed from the actual document.

Conclusion

From this project's results it is evident that the simple OCR model that uses a rudimentary form of document segmentation is highly ineffective for performing a task such as OCR. The accuracy of the EasyOCR pretrained LSTM and CNN model performs far better than the rudimentary approach for document segmentation and character recognition CNN model. Although there is a large discrepancy between the EasyOCR and simple OCR model it would be interesting to see how a better segmentation algorithm leads to slightly better results when analyzing the 3 documents in the dataset.

This opens the floor into further research investigating segmentation models for documents and how they may increase the accuracy of the OCR prediction model. In future I plan to investigate known document segmentation models such as the LSTM and other more complex ones to improve the simple OCR algorithm. My major takeaway from this project is that OCR is more complex than simple character recognition and the complexity of the problem comes from the segmentation of the document and the steps that are involved in segmenting the document in a structured manner. The journal article by Sharma further reinforces this idea by talking about other intensive techniques that may be performed to the document to produce better results in the OCR document recognition task.

Another improvement that I plan to investigate is to create another way of evaluating the OCR model. Labeling documents by hand, especially long ones, can be tedious and therefore it

would be important to research more effective ways of document labeling and accuracy metrics that would be able to tell the quality of the OCR program produced.

Overall, this project was very informative in the fact that it provided me with insight into the complexities of OCR and specifically the complexities with document segmentation. However, it is interesting to see how effective contemporary OCR models such as EasyOCR are at running OCR on scanned documents. Based on this project it is evident that OCR will certainly have its place in the future of text document processing and document automation.

Works Cited

1. Sharma, Parikshit. (2023). Advancements in OCR: A Deep Learning Algorithm for Enhanced Text Recognition. International Journal of Inventive Engineering and Sciences. 10. 1-7. 10.35940/ijies.F4263.0810823. https://www.researchgate.net/publication/373513855_Advancements_in_OCR_A_Deep_Learning_Algorithm_for_Enhanced_Text_Recognition
2. *Artificial Intelligence - A Modern Approach (3rd Edition), Stuart Russel, Peter Norvig*
3. Github Project - https://github.com/mgoldgirsh/ocr-evaluation
4. OCR-Dataset - https://www.kaggle.com/datasets/harieh/ocr-dataset
5. EasyOCR Github Repo - https://github.com/JaidedAI/EasyOCR