Use of Deep Learning for Open Format Line Detection and Handwriting Recognition: An End-to-End System

Curtis Wigington
Brigham Young University

1 Introduction

Handwriting recognition (HWR) systems have achieved low errors rates as deep learning methods have developed. A typical handwriting recognition systems consists of two parts: the segmentation and the recognition. Each part is usually trained and tuned independently. Given a sufficient amount of annotated data, both segmentations and transcriptions, the division in the systems may not be an issue. Unfortunately, because training recognition models was not considered, many collections were transcribed without recording the segmentations. This poses a problem because without segmentation annotations a segmentation system cannot be trained. Furthermore, because the HWR system requires segmented handwriting, it cannot learn from the transcription annotations alone.

We propose a HWR system that performs both the line-level segmentation and recognition in an end-to-end trainable system. A key part of our proposed method is that you only need a few documents with line level segmentations to pretrain the system and then the system trains entirely on the document transcriptions.

2 Methods

Our system consists of three parts: the start of line finder, line follower, and handwriting recognizer (Figure 1). These three parts are end-to-end differentiable allowing for errors in recognition to inform the segmentation.

2.1 Start of Line Finder

We follow and extend the technique proposed by [3]. The start of line finder makes a prediction for every 16x16 window and each prediction contains five values: x and y coordinates, scale, rotation, and confidence. Rotation is an additional term we are adding which was not included in [3]. In place of an MDLSTM network as proposed in [3], we propose to use a fully convolutional network for efficiency as we believe that the entire page context is not needed to predict the start of a line.

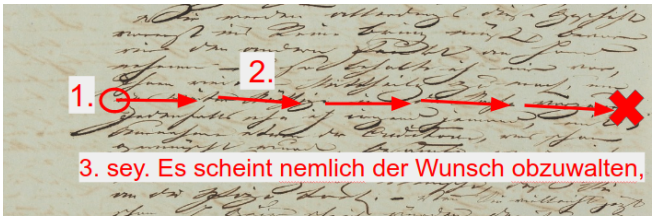

Figure 1: The three parts of the end-to-end system: (1) start of line finder, (2) line follower, and (3) handwriting recognizer.

2.2 Line Follower

The line follower is a key novel contribution of our system. The general approach of the line follower is that, given the start-of-line position, the follower extracts a small localized window based on the current position, scale, and rotation. A CNN is given the local window image as input, and regresses the next location. It also predicts a confidence value that it has not reached the end of the text line. This repeats until the network predicts that it had reached the end of the line, or it has reach a maximum number of steps. The path it followed is then segmented and passed to the handwriting recognizer.

2.3 Handwriting Recognizer

Our method is general to any line neural based recognition method that trains using CTC loss [2]. We employ a CNN-LSTM architecture. In a recent com-

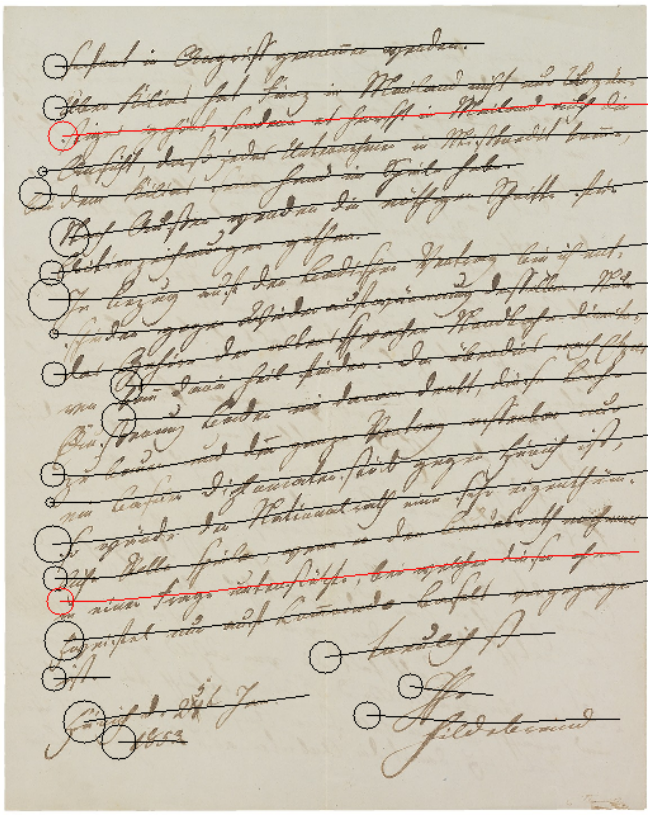

Figure 2

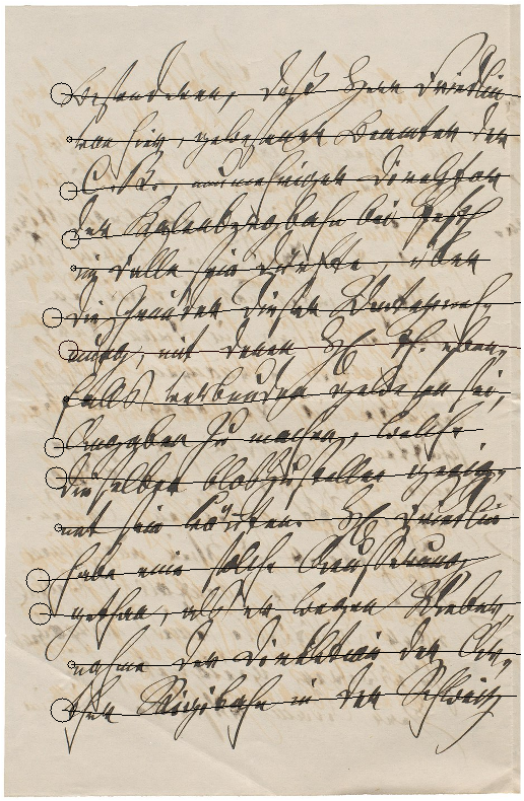

Figure 3

petition [5] CNN-LSTM methods achieved the best results. Also, a comparison of popular techniques suggests that CNN-LSTM architectures are superior [4]. We use the architecture presented in [1]

2.4 End-to-End Training

Each of these three parts are pretrained on a small collection of documents with segmentations and transcriptions. The system is trained end-to-end using only the CTC loss. This allows us to train on a much larger collection of documents.

2.5 Results

We do not currently have results on the complete end-to-end system. However, we have an intermediate result that is a significant contribution alone. Given that we have the transcription annotation, we can recover the segmentations. This done by running the start of line finder on the image and then the line follower on every predicted start of line. We then run recognition on every segmentation line. The result of the recognition network that is closest to the transcription annotation is selected (Figures 2 and 3).

References

[1] Xinyu Fu, Eugene Ch'ng, Uwe Aickelin, and Simon See. CRNN: A joint neural network for redundancy detection. *CoRR*, abs/1706.01069, 2017.

[2] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376. ACM, 2006.

[3] Bastien Moysset, Christopher Kermorvant, and Christian Wolf. Full-page text recognition: Learning where to start and when to stop. *CoRR*, abs/1704.08628, 2017.

[4] J. Puigcerver. Are multidimensional recurrent layers really necessary for handwritten text recognition? In *ICDAR 2017*.

[5] J. A. Snchez, V. Romero, A. H. Toselli, Mauricio Villegas, and E. Vidal. Icdar2017 competition on handwritten text recognition on the read dataset. In *2017 ICDAR*.