Paragraph text segmentation into lines with Recurrent Neural Networks
(2015)

A Resume

December 3, 2018

Abstract

using a more agnostic Machine Learning-based approache for text line location. To be able to process either damaged documents, or flows of documents with a high variety of layouts and other characteristics

1 Introduction

The detection of text lines is a first processing step in all text recognition systems. The level of accuracy of text boundary locations is critical for the performance of the whole system. In particular, any word missed by the text detection algorithm directly contributes to increase the lower bound of the Word Error Rate that can be expected.

2 Method

The proposed method is inspired by the latest generation of optical models used for text recognition, namely Recurrent Neural Networks. The inputs are image patches from a paragraph of a document image. Feature extraction is implicit in this 'deep model', pixel values are directly fed into the network. The network outputs predictions for a sequence of line and interline labels, which is essentially a two-class problem. The network is trained using a softmax activation function in the last layer and therefore outputs a sequence of vectors of posterior probabilities, each vector corresponding to a given patch of the input images.
RNN used scans the image vertically to detect horizontal lines of characters. In order to integrate information from an entire text line into the decision, we resort to a 2D version of these models with 2D-recurrent connections, each hidden layer is connected to two different preceedings layers, a vertical one and a horizontal one.
Four parallel LSTMs are used, one for each direction, they were trained using stochastic gradient descent on the posterior probability of the sequence, to avoid the need for exact annotation, the usage of CTC(connectionist temopral classification) aliviate the need for such annotation and only require weak annoatations, i.e. only the number of textlines for each paragraph.

3 Experimental setup

3.1 Performance metrics

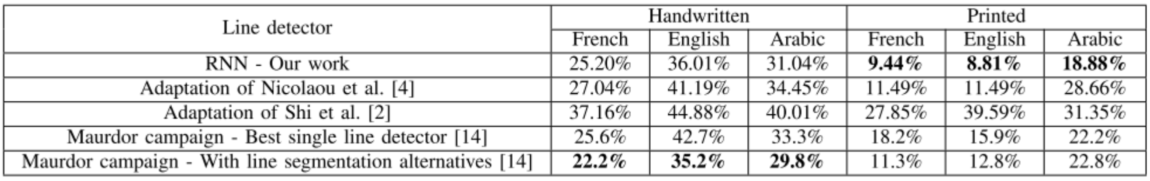
- *Error Rate in predicting the right number of lines:* The first one is the percentage of paragraphs in which the right number of lines are detected.

- *Word Error Rate (WER%):* The second one is the word error rate. The line segmentation algorithm is included in a recognition chain with another recurrent neural network as optical model for text recognition and a language model. This metric shows the improvement with respect to the real goal of the line detector algorithm, improving the text recognition.

3.2 Technical details

1. Image preprocessing, such as orientation correction, rescaling the paragraph to 300dpi, deskewing with Bloomberg's algorithm. Horizetal flip is performed for paragraphs written in Arabic to get a left alignement and the paragraph length is normelised to 200px.

2. RNN architecture, a six layer deep neural net, containing three 4-bidirectionnal LSTM layers and two convolutionnal layers, hyperbolic tagent function is used as non-linearity and dropout with a probability od 0.5 after the last LSTM layer, final layer if FCN followed by a collapse and a softmax.



Line detector	Handwritten			Printed		
	French	English	Arabic	French	English	Arabic
RNN - Our work	25.20%	36.01%	31.04%	**9.44%**	**8.81%**	**18.88%**
Adaptation of Nicolaou et al. [4]	27.04%	41.19%	34.45%	11.49%	11.49%	28.66%
Adaptation of Shi et al. [2]	37.16%	44.88%	40.01%	27.85%	39.59%	31.35%
Maurdor campaign - Best single line detector [14]	25.6%	42.7%	33.3%	18.2%	15.9%	22.2%
Maurdor campaign - With line segmentation alternatives [14]	**22.2%**	**35.2%**	**29.8%**	11.3%	12.8%	22.8%

Figure 1: Comparison between the word error rates obtained with several line segmentation algorithms on Maurdor test set

References

[1] Moysset, Bastien Kermorvant, Christopher Wolf, Christian Louradour, Jerome, *Paragraph text segmentation into lines with Recurrent Neural Networks*