Joint Line Segmentation and Transcription for End-to-End Handwritten Paragraph Recognition
Théodore Bluche

Introduction
- Handwriting recognition evolved from isolated character recognition to word recognition with explicit segmentation to complete line recognition with implicit character segmentation.
- Nowadays handwriting recognition systems still need cropped text lines for both training and transcription.
- The recent works on attention models showed that it was possible to learn to align and translate (Bahdanau et al., 2014), or describe (Xu et al. 2015)

Handwritten Text Recognition with MDLSTM (Graves et al., 2008)
- Text line images are fed to a Multi-Dimensional LSTM layer
- Feature maps are transformed and subsampled by convolutional layers
- This is repeated several times.
- At the end, there is one feature map per character
- They are collapsed in the vertical direction to obtain sequences of character predictions

Proposed Solution
Standard Collapse
Simple vertical sum of features (all have the same importance)
Same error propagated across vertical positions
Prevents the recognition of several text lines

Weighted Collapse
$z_j = \sum_i f_{ij}$
$z_j^{(t)} = \sum_i \omega_{ij}^{(t)} f_{ij}$
We propose to multiply the feature maps by a map of weights

A neural network predicts a score for each position + softmax on columns

Applied several times, we can iteratively focus on the successive text lines in a paragraph

Neural Network
- We keep the MDLSTM network (before collapse) as an encoder of the input paragraph image
- The attention network is applied iteratively on the feature maps and performs an implicit line segmentation
- The obtained sequences are concatenated and fed to a bi-directional LSTM decoder

Quantitative Results
Outperforms line-by-line recognition with explicit line segmentation (ground-truth positions or automatic algorithms, Char.Error.Rate )

<table>
<thead>
<tr>
<th>Database</th>
<th>Resolution</th>
<th>GroundTruth</th>
<th>Projection</th>
<th>Shredding</th>
<th>Energy</th>
<th>This work</th>
</tr>
</thead>
<tbody>
<tr>
<td>IAM</td>
<td>150 dpi</td>
<td>8.4</td>
<td>15.5</td>
<td>9.3</td>
<td>10.2</td>
<td>6.8</td>
</tr>
<tr>
<td></td>
<td>300 dpi</td>
<td>6.6</td>
<td>13.8</td>
<td>7.5</td>
<td>7.9</td>
<td>4.9</td>
</tr>
<tr>
<td>Rimes</td>
<td>150 dpi</td>
<td>4.8</td>
<td>6.3</td>
<td>2.9</td>
<td>8.2</td>
<td>2.5</td>
</tr>
<tr>
<td></td>
<td>300 dpi</td>
<td>3.6</td>
<td>5.0</td>
<td>4.5</td>
<td>6.6</td>
<td>2.5</td>
</tr>
</tbody>
</table>

Processing time comparable to segment + reco, and much faster than the “Scan, Attend and Read” model (Bluche et al., 2016)

The results are competitive with the state-of-the-art (which uses ground-truth text-line positions)

<table>
<thead>
<tr>
<th>Method</th>
<th>Processing time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GroundTruth (crop+reco)</td>
<td>0.21 ± 0.07</td>
</tr>
<tr>
<td>Shredding (segment+reco)</td>
<td>0.75 ± 0.26</td>
</tr>
<tr>
<td>Scan, Attend and Read (reco)</td>
<td>21.2 ± 5.6</td>
</tr>
<tr>
<td>This Work (reco)</td>
<td>0.62 ± 0.14</td>
</tr>
</tbody>
</table>

Conclusions
We proposed a neural network for end-to-end transcription of handwritten paragraphs
An implicit line segmentation is performed by the network, which we can train directly at the paragraph level (no need for line-level positions and annotations)
Limitation: the attention spans the whole width of the image
=> need to refine the attention mechanism to handle arbitrary documents.