Connectionist Temporal Classification: Labelling Unsegmented Sequences with Recurrent Neural Networks

Thomas Mesnard, Alex Auvolat
Probabilistic Graphical Models Project, MVA Master

Abstract
Many real-world sequence learning tasks require the prediction of sequences of labels from noisy, unsegmented input data. Recurrent neural networks (RNNs) are powerful sequence learners that would seem well suited to such tasks. However, because they require pre-segmented training data, and post-processing to transform their outputs into label sequences, they cannot be applied directly. CTC is a method for training RNNs to label unsegmented sequences directly, thereby solving both problems.

Main Idea
RNNs are powerful learners for sequences, but:
- Standard methods need pre-segmented training data
- Need for complex post-processing
CTC solves this problem:
- Able to train RNNs using unsegmented training data
- Learns the segmentation automatically
- Provides directly usable output
This method is now extremely used, even by Google!

Recurrence equations
We define the following notation:
- \( y_t^k \): output at time \( t \) for symbol \( k \)
- \( l_t \): label
- \( \alpha_t(s) \): label with blanks
Initialisation:
- \( \alpha_1(1) = y_1^k \)
- \( \alpha_1(2) = y_1^k \)
- \( \alpha_1(s) = 0, \forall s > 2 \)
Recurrence relation:
- \( \alpha_{t+1}(s) = \begin{cases} \alpha_t(s)y_t^k & \text{if } l_t = b \text{ or } l_{t-2} = l_t' \\ (\alpha_t(s) + \alpha_{t-1}(s-2))y_t^k & \text{otherwise} \end{cases} \)
Finally, we have:
- \( p(l|x) = \alpha_T(l'|l) + \alpha_T(l'|l - 1) \)

Toy dataset
We first tried our implementation on a simple task:
1. 2345
2. 321
3. 5432
4. 432
5. 543

• A RNN can easily solve this
• It needs to read the full sequence before predicting a label
• CTC provides satisfactory results

Results

<table>
<thead>
<tr>
<th>Train/Valid</th>
<th>Error rate</th>
<th>Mean edit distance</th>
<th>Errors per character</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>0.067</td>
<td>1.0</td>
<td>0.06</td>
</tr>
<tr>
<td>Valid</td>
<td>0.063</td>
<td>1.1</td>
<td>0.09</td>
</tr>
</tbody>
</table>

Table 1: Performances of CTC on our toy dataset

Conclusion
CTC is a very powerful model, and also has a nice mathematical formulation. It is also very used in practice (most successfull applications: speech recognition, handwriting recognition).

Contact Information
- Web: http://github.com/thomasmesnard/CTC-LSTM
- Email: thomas.mesnard@ ens.fr
- Alex.auvolat@ ens.fr

References