The task:
Detect errors in learner writing (spelling, grammar, word usage, etc).

Examples:
I want to travel on July and because of it is more suitable for me.
We don’t need to wear clothes like layer and layer.
The restaurant was closed because unknown reasons.

Applications:
• Immediate feedback in self-tutoring systems for language learning.
• Automated exam grading for language testing.
• Providing language checking in general writing applications.

Compositional Architectures
Experimenting with alternative architectures for error detection.

• Convolutional network with window size 7 around the target word.
• Deep convolutional network, using an extra convolution to capture higher-order features.
• Bidirectional RNN, constructing context representations with Elman-style RNNs.
• Deep bidirectional RNN, using two stacked layers of RNNs.
• Bidirectional LSTM, allowing the recurrent component to select which context to keep.
• Deep bidirectional LSTM, adding a second LSTM layer.
• For comparison, a Conditional Random Fields (CRF, Lafferty et al., 2001) model, as implemented in CRF++.

Experiments

Models evaluated by detecting errors in the publicly released FCE dataset of learner writing (Yannakoudakis et al., 2011).
The best results for error detection were achieved with a bidirectional LSTM architecture, using pretrained embeddings, an extra narrow hidden layer, and a softmax output layer.

<table>
<thead>
<tr>
<th>Development</th>
<th>P</th>
<th>R</th>
<th>F0.5</th>
<th>Test</th>
<th>P</th>
<th>R</th>
<th>F0.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRF</td>
<td>62.2</td>
<td>13.6</td>
<td>36.3</td>
<td>56.5</td>
<td>8.2</td>
<td>25.9</td>
<td></td>
</tr>
<tr>
<td>CNN</td>
<td>52.4</td>
<td>24.9</td>
<td>42.9</td>
<td>46.0</td>
<td>25.7</td>
<td>39.8</td>
<td></td>
</tr>
<tr>
<td>Bi-RNN</td>
<td>63.9</td>
<td>18.0</td>
<td>42.3</td>
<td>51.3</td>
<td>19.0</td>
<td>38.2</td>
<td></td>
</tr>
<tr>
<td>Bi-LSTM</td>
<td>54.5</td>
<td>28.2</td>
<td>46.0</td>
<td>46.1</td>
<td>28.5</td>
<td>41.1</td>
<td></td>
</tr>
<tr>
<td>Deep Bi-LSTM</td>
<td>56.7</td>
<td>21.3</td>
<td>42.5</td>
<td>48.2</td>
<td>21.6</td>
<td>38.6</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Performance of alternative models on the FCE dataset.

Additional Training Data
We found that error detection results could be substantially improved by using additional training data.
Including NUCLE had almost no effect on performance, likely due to differences in writing requirements.
The network performance plateaued around 8M tokens of training data.

CoNLL-14 Shared Task Dataset
CoNLL-14 error correction dataset (Ng et al., 2014) converted to an error detection task.
The network outperformed all shared task systems, with an absolute improvement of 3%, without using manual engineering.

<table>
<thead>
<tr>
<th>Human annotator</th>
<th>Bi-LSTM (full)</th>
<th>CAMB</th>
<th>CULP</th>
<th>AMU</th>
<th>Bi-LSTM (FCE-public)</th>
</tr>
</thead>
<tbody>
<tr>
<td>79.6</td>
<td>76.0</td>
<td>76.0</td>
<td>78.0</td>
<td>79.9</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Pearson’s correlation $r$ and Spearman’s correlation $\rho$ on essay scoring.

Essay Scoring
We integrated probabilities from the error detection system as features in an essay scoring system.
Achieved substantial improvements over state-of-the-art and performance comparable to human annotators.

Table 2: Pearson’s correlation $r$ and Spearman’s correlation $\rho$ on essay scoring.

Example Output
The main events of the party will end up at about 12:30 in the night.
Or even in cars and washmachines there’re computer chips.
Finally, the last day I suggest you to go to the mall where you can enjoy shopping and looking around.
Your hotel is called Palace Hotel and it is placed in the city centre.