Compositional Sequence Labeling Models for Error Detection in Learner Writing

Marek Rei and Helen Yannakoudakis
University of Cambridge

Error Detection

The task:

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

Examples:

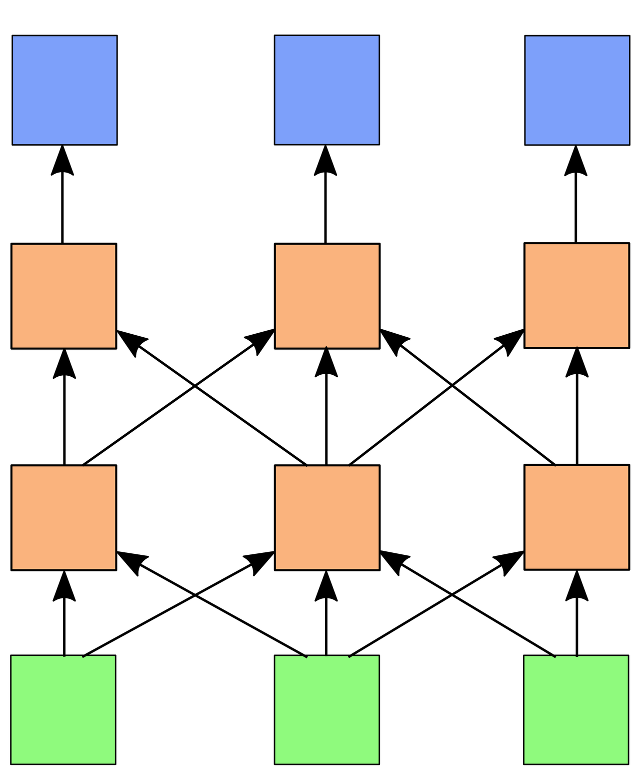
I want to travel ***on*** *July and because* ***of*** *it is more suitable for me.*
We don't need to wear clothes ***like layer and layer****.*
The restaurant was closed because ***unknown*** *reasons.*

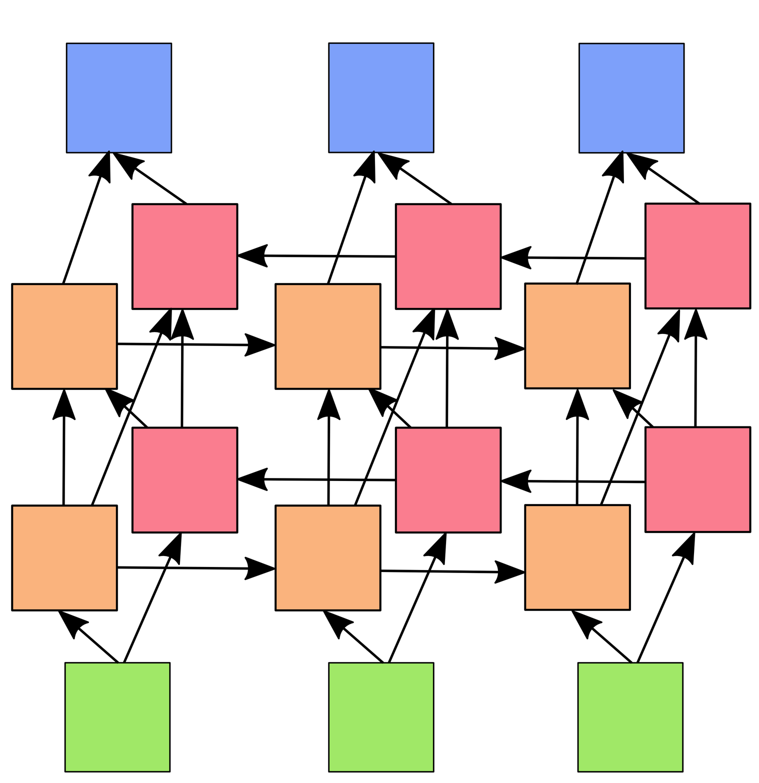
Applications:

- 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.

	Development			Test		
	P	R	$F_{0.5}$	P	R	$F_{0.5}$
CRF	62.2	13.6	36.3	56.5	8.2	25.9
CNN	52.4	24.9	42.9	46.0	25.7	39.8
Bi-RNN	**63.9**	18.0	42.3	**51.3**	19.0	38.2
Bi-LSTM	54.5	**28.2**	**46.0**	46.1	**28.5**	**41.1**
Deep Bi-LSTM	56.7	21.3	42.5	48.2	21.6	38.6

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.

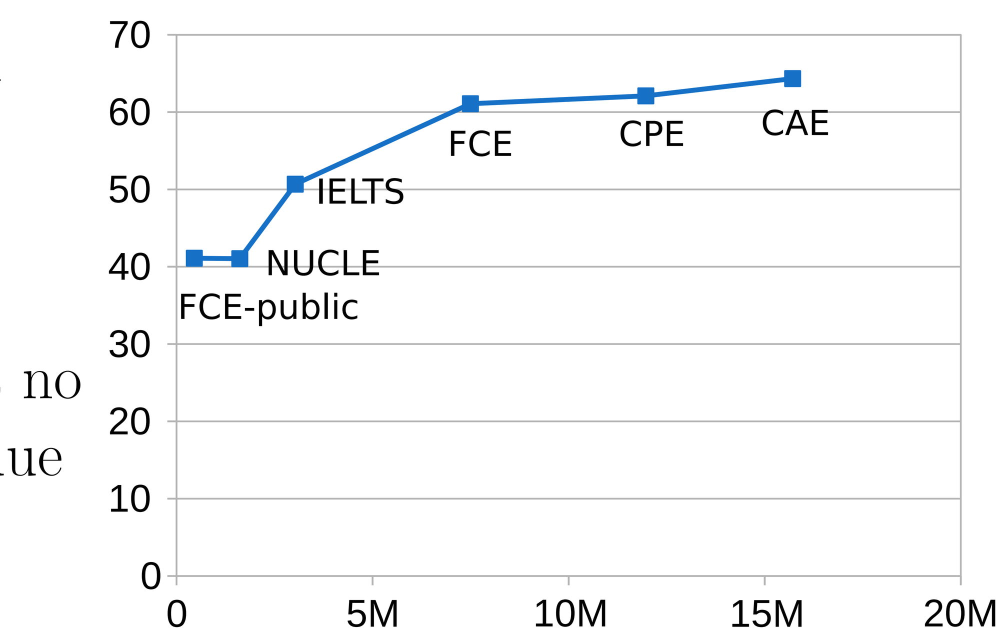


Figure 1: $F_{0.5}$ measure on the public FCE test set, as a function of the total number of tokens in the training set.

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.

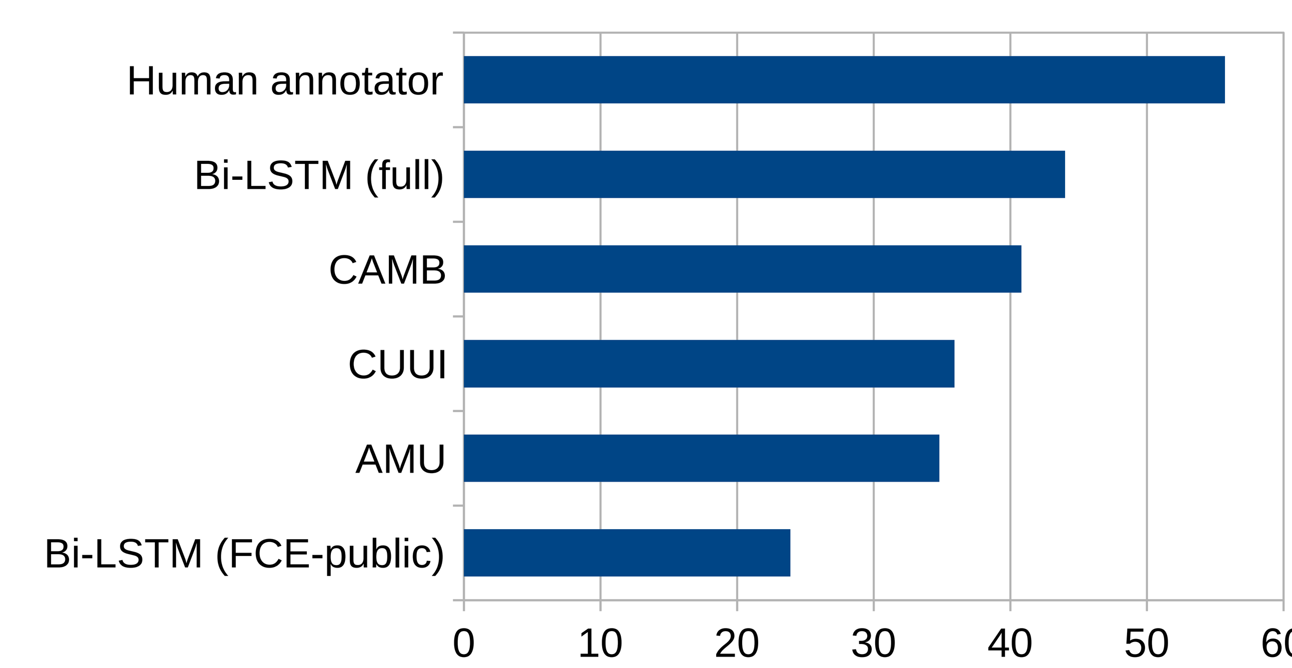


Figure 2: $F_{0.5}$ detection score on the CoNLL-14 Shared Task dataset (annotation 2).

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.

	r	ρ
Human annotators	79.6	79.2
SAT	75.1	76.0
SAT + Bi-LSTM (FCE-public)	76.0	77.0
SAT + Bi-LSTM (full)	**78.0**	**79.9**

Table 2: Pearson's correlation r and Spearman's correlation ρ 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 **sugget** 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.