Automatic Text Scoring Using Neural Networks

Dimitrios Alikaniotis [1] Helen Yannakoudakis [2] Marek Rei [2]
[1] Department of Theoretical and Applied Linguistics
[2] ALTA Institute, Computer Laboratory, University of Cambridge

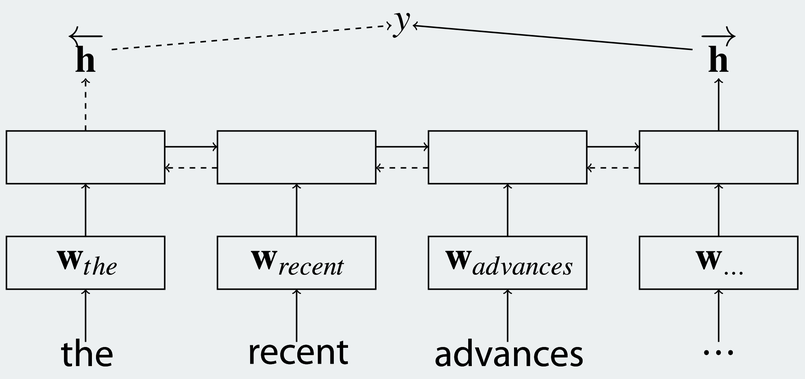
UNIVERSITY OF CAMBRIDGE

Motivation

- Current approaches to text scoring [4] use rich linguistic features to capture the aspects of writing to be assessed.
- However, substantial effort is needed from experts to hand-select and tune those features for specific domains.
- Deep-learning systems do not need any manual feature engineering and have been shown to surpass many state-of-the-art models in NLP tasks [1].
- The downside is that their marking criteria cannot be directly interpreted.
- We, therefore, propose (1) a novel way to automatically extract features from the texts using deep learning techniques and (2) a method to visualize what the model learns.

Score-Specific Word Embeddings (SSWE)

- Most current approaches [2, 3] to building word embeddings capture only contextual information for each word.
- We extend this approach to capture both **contextual** and **usage** information for words.
- For contextual information, the task of the network is to distinguish between **true** and **corrupt** n-grams (i.e., `the cat sat' $>$ `the mat sat').
- For the usage part, the network predicts the essay score from each word using linear regression.
- The resulting word embeddings carry information about (1) the contexts in which the word appears, and (2) the likely score each word may take.
- We form **essay** embeddings using the word embeddings as input to a Long-Short Term Memory (LSTM) network.
- The activation of the hidden layer at the last timestep is used to predict the essay score using linear regression.
- We explore different combinations of bi-directional and multi-layer neural networks. For the bi-directional models we concatenate the two hidden layers to predict the essay score. During training, we continue tuning the word embeddings by propagating the error gradients back to them.

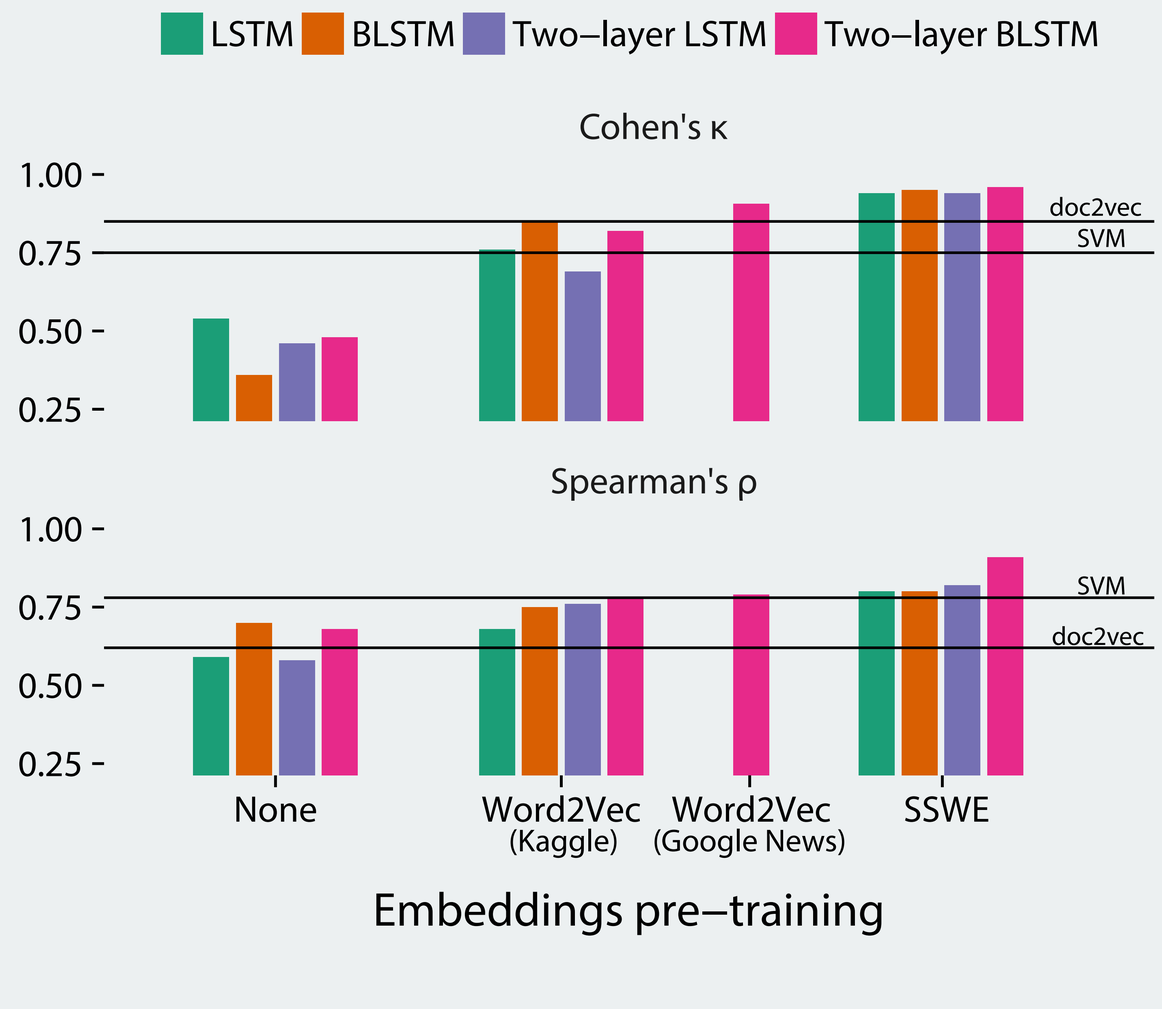


Other Baselines

- We compare the SSWE method against different methods of extracting word and document embeddings.
- We also compare our results to more traditional models using rich linguistic features.
- We construct word embeddings using:
  - `word2vec` embeddings tuned on our training set
  - `word2vec` embeddings trained on the Google News corpus
  - Embeddings, which are constructed on the fly by the LSTM
- We construct document embeddings using:
  - `doc2vec` paragraph embeddings for each essay
- We also train a Support Vector Regression model on manually engineered features, such as character and part-of-speech unigrams, bigrams and trigrams; word unigrams, bigrams and trigrams and the distribution of common nouns, prepositions, and coordinators.

Results

- We trained the models on the Kaggle dataset (ca. 12K texts) and we report the coefficients between our predicted scores and the gold standard on a seperate testing set (64% training, 20% testing and 16% validation).
- Datasets are released for future comparison.



Visualization

- We cannot infer directly the marking criteria the LSTM is using to predict the scores. We can, however, see how much the model prefers certain words. If an embedding does not change much when predicting a low scoring essay that indicates a low-quality word. Conversely, if an embedding does not change much when predicting a high scoring essay that indicates a high-quality word.
- We find the quality of the embeddings by **tricking** the network.
- Without updating the weights we record the error gradients when feeding an essay along with (1) the lowest and (2) the highest possible score.
- We find how much an embedding needs to change by taking the **magnitude** of the Jacobian.

…is in this picture the cyclist is riding a dry and area which could mean that it is very and the looks to be going down hill there looks to be a lot of turns . …

…The only reason im putting this in my own way is because know one is patient in my family . …

… Whether they are building hand-eye coordination , researching a country , or family and friends through @CAPS3 , @CAPS2 , @CAPS6 the internet is highly and I hope you feel the same way .

green = high quality vectors; red = low quality vectors

References

[1] Ciprian Chelba, Tomáš Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. One Billion Word Benchmark for Measuring Progress in Statistical Language Modeling. In *arXiv preprint*, 2013. URL `http://arxiv.org/abs/1312.3005`.

[2] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. *Proceedings of the 25th international conference on Machine learning*, pages 160--167, 2008.

[3] Tomas Mikolov, Greg Corrado, Kai Chen, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. *Proceedings of the International Conference on Learning Representations (ICLR 2013)*, pages 1--12, 2013. URL `http://arxiv.org/pdf/1301.3781v3.pdf`.

[4] Mark Shermis and Ben Hammer. Contrasting state-of-the-art automated scoring of essays: analysis. Technical report, The University of Akron and Kaggle, 2012.