Sentence Unscrambler: Exploring Deep Learning Models for Word Linearization

CS224N Natural Language Processing with Deep Learning

Kent Vainio – kentv@stanford.edu, Jason Zheng – jzzheng@stanford.edu, Sonja Johnson-Yu – sonjyu@stanford.edu

Overview

Linearization: given a bag of words, order them into a grammatical sentence.

- Traditional approach uses statistical models
- Recent approaches use LSTMs [1]
  - With or without syntactic linearization (building syntax trees) [2]
- Syntax-free linearizer avoids parsing error and is more lightweight

Project Goal: Improve syntax-free neural linearizer using encoders and attention.

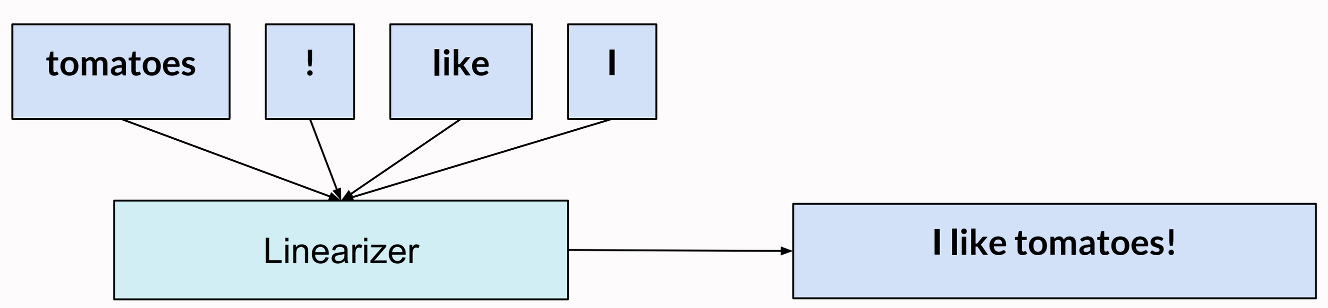

Figure 1. An overview of the task of linearization

Dataset and Approach

1) Dataset = three NLTK corpora

- Gutenberg, Brown, Reuters
- multiple genres & time periods
- omit sentences with > 20 tokens
- 96,805 sentences
- dataset sizes:
  - 1000/10,000/96,805

2) Input Generation

- Split into tokens
  - words + punctuation
- Randomize order

3) Run through model

- embedding lookup
- optional encoder
  - with or without attention
- decoder
  - greedy or beam search
  - with or without random <unk> replacement

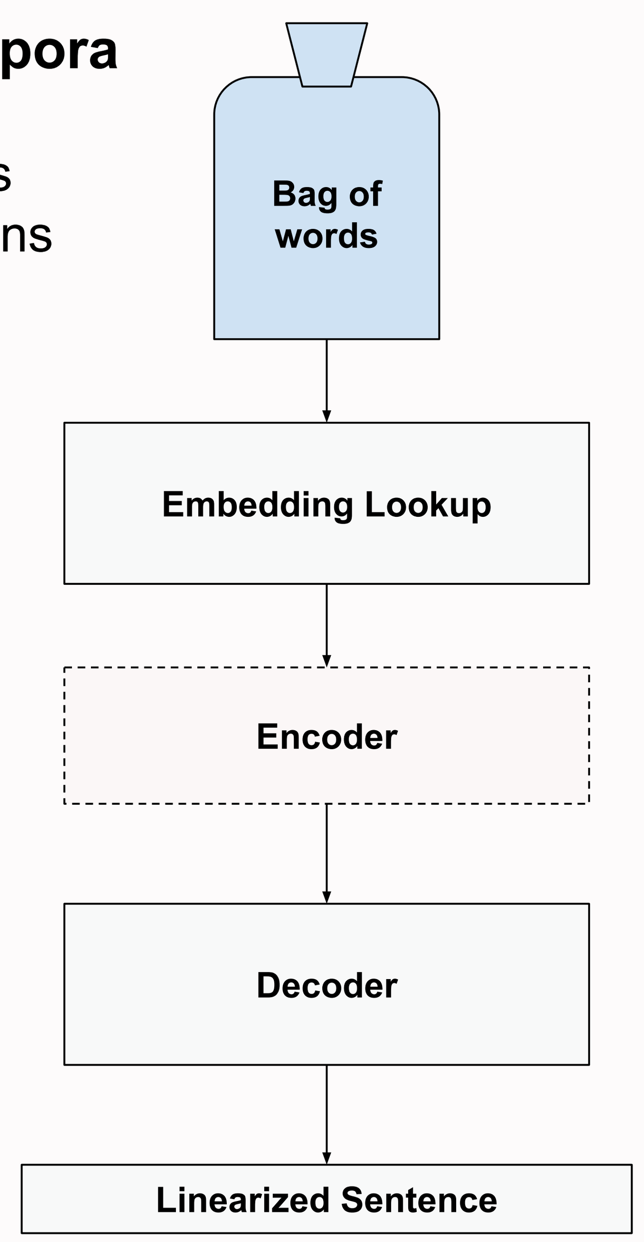

Figure 2. Linearization Model Overview

Results and Analysis

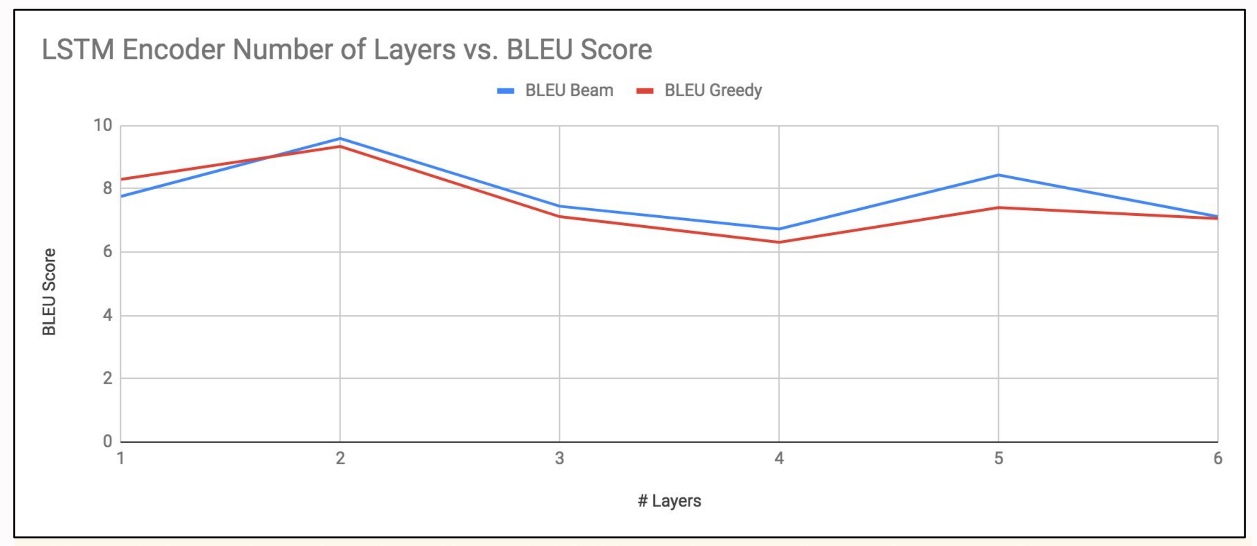

Figure 4. Comparison of number of layers in LSTM encoder, as well as performance of beam vs. greedy search, on 1000 samples. 2 layers is best, as is beam search.

Experiments:

- baseline LSTM
- *n*-layer bidirectional LSTM encoder
- *n*-layer CNN encoder
- greedy vs. beam search
- w/ vs. w/o <unk> replacement
- w/ vs. w/o attention
- w/ vs. w/o highway layer

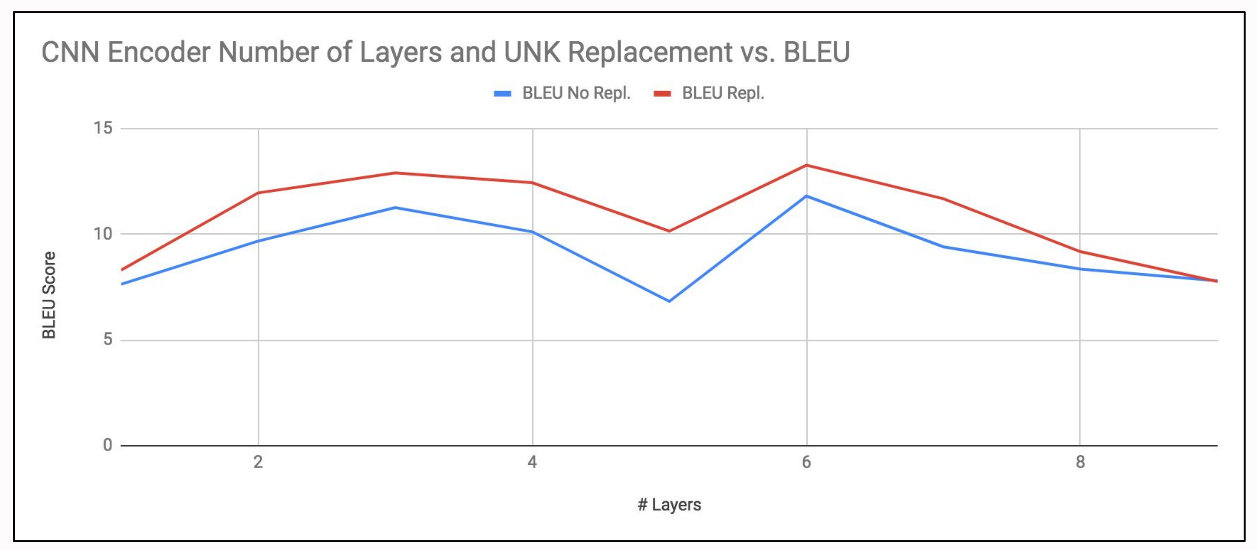

Figure 5. Comparison of number of layers in CNN encoder, as well as performance of UNK replacement vs. none, on 1000 samples

Optimal # of Layers:

- LSTM: 2
- CNN: 3

Follow-up Experiments:

(5 trials on 970 samples)

- CNN Highway: 6.57
- CNN No Highway: 7.51

(1 trial on 9700 samples)

- CNN-3: 14.18
- CNN-6: 12.85

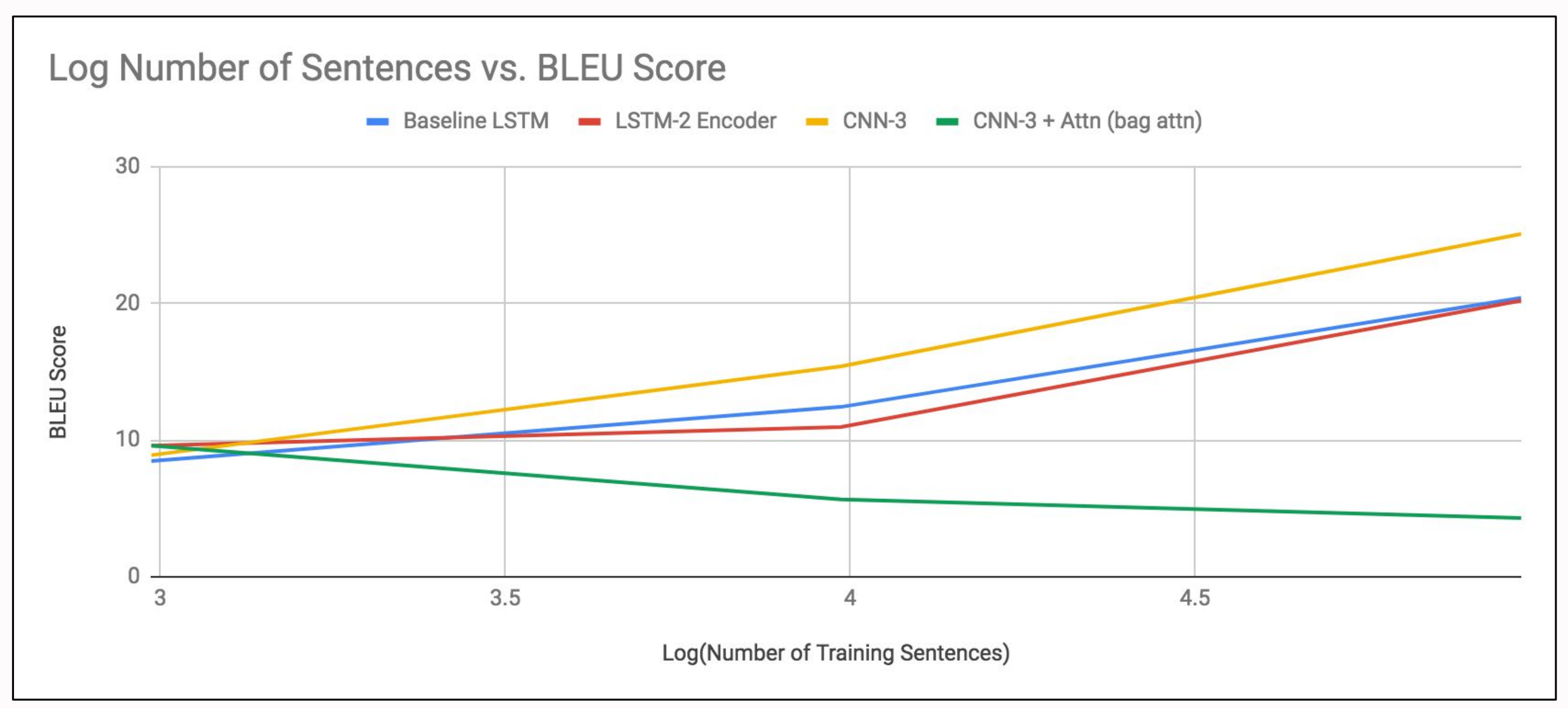

Figure 6. Comparison of different models on datasets of varying sizes. CNN-3 without attention performs best.

Summary

- CNN-3 yields highest BLEU scores
- Attention leads to poorer performance
- LSTM encoder performs similarly to baseline

Qualitative Analysis

Baseline	CNN-3	Reference	Evaluation
He was that . its ceiling denied production Opec exceeding agreed	He denied that Opec was exceeding its agreed production ceiling .	He denied that Opec was exceeding its agreed production ceiling .	Perfect
The two of three , and children . 2 : 4 hundred seventy Shephatiah	2 : 4 The children of Shephatiah , three hundred seventy and two .	2 : 4 The children of Shephatiah , three hundred seventy and two .	Perfect
It said the new process , xylene and xylene , include isomerization hydrodealkylation . units fractionation extraction thermal aromatic BTX	It said the new units and include hydrodealkylation , isomerization , xylene xylene . extraction process thermal aromatic fractionation BTX	It said the new BTX process units include aromatic extraction , xylene fractionation , xylene isomerization and thermal hydrodealkylation .	Bad, <unk> problem

Figure 7. Outputs of baseline and CNN-3, in comparison to reference sentences. The CNN-3 notably outperforms the baseline.

Conclusion

- 3-Layer CNN Encoder performs best
- **Improves on baseline by ~4.5 BLEU points**
- LSTM Encoder performs similarly to baseline
- UNK replacement yields higher BLEU score
- Beam search yields higher BLEU score
- Attention decreases BLEU score on full dataset
- Challenges for the model:
  - rare vocabulary
  - very long sentences

Experimental Model Summary

Data Size	Baseline LSTM	LSTM-2 Encoder	CNN-3 Encoder	CNN-3 Encoder + Bag Attention
Small	8.46	**9.59**	8.89	**9.59**
Med	12.42	10.95	**15.38**	5.65
Full	20.4	20.19	**25.06**	4.29

Figure 8. Comparison of different models on datasets of varying sizes. Bolded are the models that performs best for given dataset size.

Future Work

- Char-LSTM for handling <unk>s
- Transformer model
- Pointer-generator networks

References

[1] Alexander M. Rush, Allen Schmaltz, and Stuart Shieber. Word ordering without syntax. *Conference on Empirical Methods in Natural Language Processing(EMNLP-16). Austin, Texas*, pages 2319–2324, 2016.

[2] Yue Zhang, Linfeng Song, and Daniel Gildea. Neural transition-based syntactic linearization. *INLG 2018 (International Natural Language Generation Conference). Tilburg, Netherlands*, 2018.