Stanford University

Sentence Simplification with Transformer-XL and Paraphrase Rules

Fei Fang, Matthew Stevens

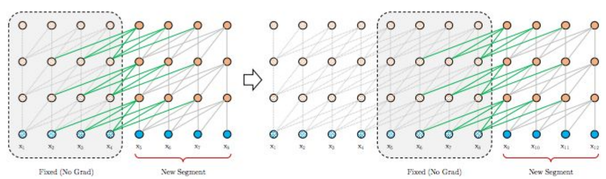
Problem

Given a sentence with complex syntax and/or diction as input, a sentence simplification model aims to output a sentence that is easier to comprehend and yet preserves the semantics of the input. This technology helps readers process and retain information more efficiently. Furthermore, it is especially useful to children, nonnative readers, as well as those with reading disabilities such as dyslexia.

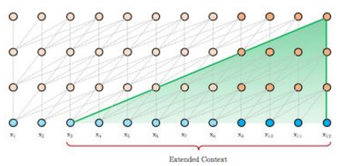
A variety of statistical systems as well as neural architectures have been applied to this task, while the most recent SOTA results have been achieved with a hybrid of the two approaches. Thus, in this project, we decided to investigate the potential of hybrid neural-statistical models for sentence simplification.

Model

Transformer-XL: Transformer with recurrence



(a) Training phase.



(b) Evaluation phase.

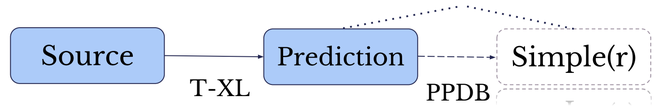
A context is split into segments, and each segment is processed by a Transformer. When a segment is processed, each hidden layer of the current Transformer receives outputs from both the current Transformer (gray arrows) and the Transformer of the previous segment (green arrows).

Source: Dai et al., "Transformer-xl: Attentive language models beyond a fixed-length context." 2019.

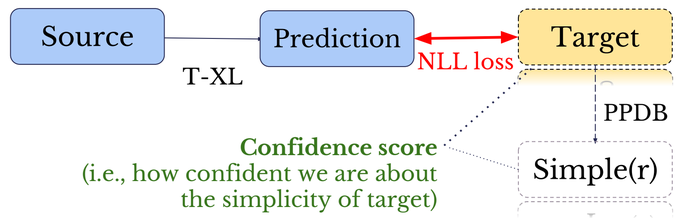
Approach

Incorporating paraphrase rules in loss functions
In our loss computations, we incorporated the paraphrase rules from the Simple Paraphrase Database (PPDB).

- **ModificationLoss** := **cosine distance**



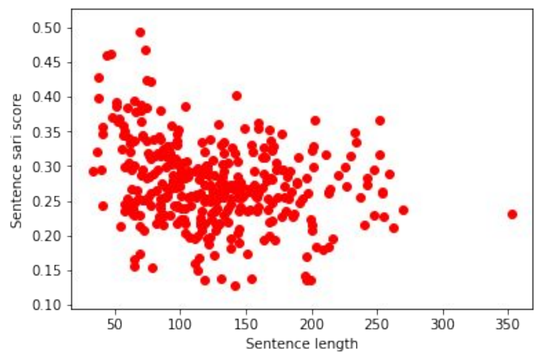
- **ConfidenceLoss** := NLL loss × **Confidence score**



Results & Analysis

Model	SARI (Eval Metric)
Baseline (LSTM Enc-Dec w/ Attention)	13.61
SOTA (Statistical model)	**30.46**
Transformer-XL w/ character embeddings	27.39

SARI: Compares predictions against human references; rewards keeping, deleting, or adding words that humans also keep, delete, or add. Not entirely appropriate for char-level models.



Data & Task

Task: Given a complex sentence $I = (x_1, x_2, \ldots, x_n)$ as input, the model outputs a simplified sentence, $O = (y_1, y_2, \ldots, y_m)$

Data:
Train: *WikiLarge*, a parallel English-Simple English Wikipedia corpus of 296,402 complex-simple sentence pairs.

Dev & test: *WikiSmall-AMT*, a corpus of 2,359 sentences, each with 8 references collected from Amazon Mechanical Turk (dev set size of 2,000 and test set size of 359).

Conclusions

- Implemented a Transformer-XL-based model for sentence simplification and investigated two methods of integrating paraphrase rules
- First application of the Transformer-XL architecture to a non-LM task, as well as the first study on character-level sentence simplification
- With some more sophisticated adaptation of Transformer-XL, the architecture has tremendous potential in tasks including but not limited to sentence simplification
- Future avenues of research include improving the SimplePPDB for use in real-world sentence simplification tasks and exploring the efficacy of weighting losses based on target sentence quality using larger datasets

References:
[1] Dai et al., "Transformer-xl: Attentive language models beyond a fixed-length context." Jan 2019.
[2] Zhao et al., "Integrating Transformer and Paraphrase Rules for Sentence Simplification." Jan 2019.
[3] Zhang and Lapata, "Sentence Simplification with Deep Reinforcement Learning." July 2017.