Does Data Augmentation Matter More Than Architecture Design for Small Datasets?
Max Sobol Mark | maxsobolmark@stanford.edu
Computer Science Department, Stanford University

Motivation

Transformer-XL is a method that combines the best of Attention-based methods and RNNs like LSTMs. It is able to model very long-distance relationships between tokens in the input text efficiently.

Reproducibility concerns: Without mentioning it in their paper, they use data-augmentation techniques that are essential to getting good performance in the small data regime. People who tried implementing their approach created a bounty for whoever was able to match LSTM performance with Transformer-XL.

Objective: Answer the question: Are data-augmentation techniques more influential than model architecture for small datasets?

Problem Setting - SQuAD

Example from the dataset:

Question: Why was Tesla returned to Gospic?

Context paragraph: On 24 March 1879, Tesla was returned to Gospic under police guard for not having a residence permit. On 17 April 1879, Milutin Tesla died at the age of 60 after contracting an unspecified illness (although some sources say that he died of a stroke). During that year, Tesla taught a large class of students in his old school, Higher Real Gymnasium, in Gospic.

Answer: not having a residence permit

Methods - Transformer-XL architecture

- In the vanilla Transformer architecture, the self-attention mechanism can only pay attention to the current context of tokens.
- The Transformer-XL has a segment-level recurrence with state reuse. This allows it to look arbitrarily back in the input text.
- To differentiate between tokens of the current segment and the previous segment, they use Relative Positional Encodings. It adds a learnable vector that encodes the position of tokens.
- Our model outputs the start and end indices of the predicted answer

Results

![Figure 1: Training loss curves for Transformer-XL models. Grey is the baseline model without dropout. Green is the word embedding dropout model. Pink is the discrete embedding dropout variation.](image)

- Transformer-XL No Dropout: F1: 12.80, EM: 10.87
- Transformer-XL Discrete Dropout 0.1: F1: 03.37, EM: 01.23
- Transformer-XL Standard Embedding Dropout 0.1: F1: 06.78, EM: 04.54
- BiDAF Discrete Dropout 0.1: F1: 60.4, EM: 57.22
- Baseline (BiDAF No Dropout): F1: 58, EM: 55

Discussion

- Transformer-XL models were not trained for enough time to reach convergence, and they take much longer to train than LSTM-based models. This is why we are seeing such poor results.
- Using discrete dropout does improve the performance of our LSTM-based model

Next steps

- The immediate next step is to train our Transformer-XL models for longer to reach convergence and do fair comparisons with the LSTM baseline.

130k training samples
6k dev samples
6k test samples

Dropout techniques

1. Discrete Embedding Dropout
   Drops entire word embeddings with some probability.
2. Standard dropout for input embeddings
   Drops out elements of the embeddings randomly.

They act as data-augmentation techniques, similar to adding noise to images. They improve generalization, but require more time to train.