Reproducing InferSent: summary of findings

Blazej Manczak

April 2021

I've trained the four models on the SNLI dataset. Those four models differed in the type of sentence encoders:

1. Baseline: averaging GloVe (840B) word embeddings.
2. Uni-directional LSTM, last hidden state being the sentence representation
3. Bi-directional LSTM with the concatenation of last hidden states being the sentence representation
4. Bi-directional LSTM with max-pooling over the concatenation of the hidden states being the sentence representation

All algorithms were trained with an SGD optimizer with a learning curriculum as described in the original paper. The training has been stopped if the model did not improve on the validation set for 3 consecutive epochs. Each sentence encoder was followed by a hidden layer with dimension 512. All the models **except** model 4 were trained with ReLu non-linearities before and after the hidden layer. Model 4 on SNLI with ReLU only marginally improved accuracy on SNLI (no time to test on SentEval). Without non-linearities, the improvement was significantly larger. Please find the results for both SNLI and SentEval summarized in Table 1.

Model	Dim	NLI test	NLI val	Transfer micro	Transfer macro
AWE	300	64.73	64.15	84.12	79.79
Uni-LSTM	2048	81.11	81.52	83.38	80.05
Bi-LSTM	4096	80.60	81.10	85.94	82.63
Bi-LSTM Max	4096	82.56	82.40	86.90	83.82

Table 1: Performance of different models.

Somewhat unexpectedly we see that just averaging the GloVe word embeddings is a very strong baseline, performing similarly to the uni-directional LSTM. One should not that the difference in accuracies for SNLI is much larger. It shows that these highly parameterized models capture not only general-purpose sentence representation but also utilize some biases and artifacts of the dataset.

One can see that for NLI the test accuracies are very close to the validation results. This might be caused by conservative early stopping. Looking at the training plots we see that training for a couple more epochs might have proven beneficial for the SNLI dataset. However, this early stopping seems to benefit the performance on the transfer tasks.

I've also investigated the impact of length of premise/hypothesis on the models' performance. I first calculated the 10% and 90% percentile of the length of hypothesis and premise (see demoNotebook). Then I've selected the corresponding examples in the test set and evaluated the model on them. The results can be found in Figure 1.

For all models except the bi-LSTM max pool, we see the general trend: for longer sentences, especially hypothesis, the models perform worse. This is because long-distance relationships are harder to encode and possibly because the longer sentences can be more convoluted.

As expected, we see that the uni-directional LSTM performs worse than the bi-directional counterparts due to the increased capacity of encoding longer sequences. We also see that the ability to make a "sharp" decision in bi-LSTM max-pooling models mitigates the problem present in other models.

The code repositiory can be found here.

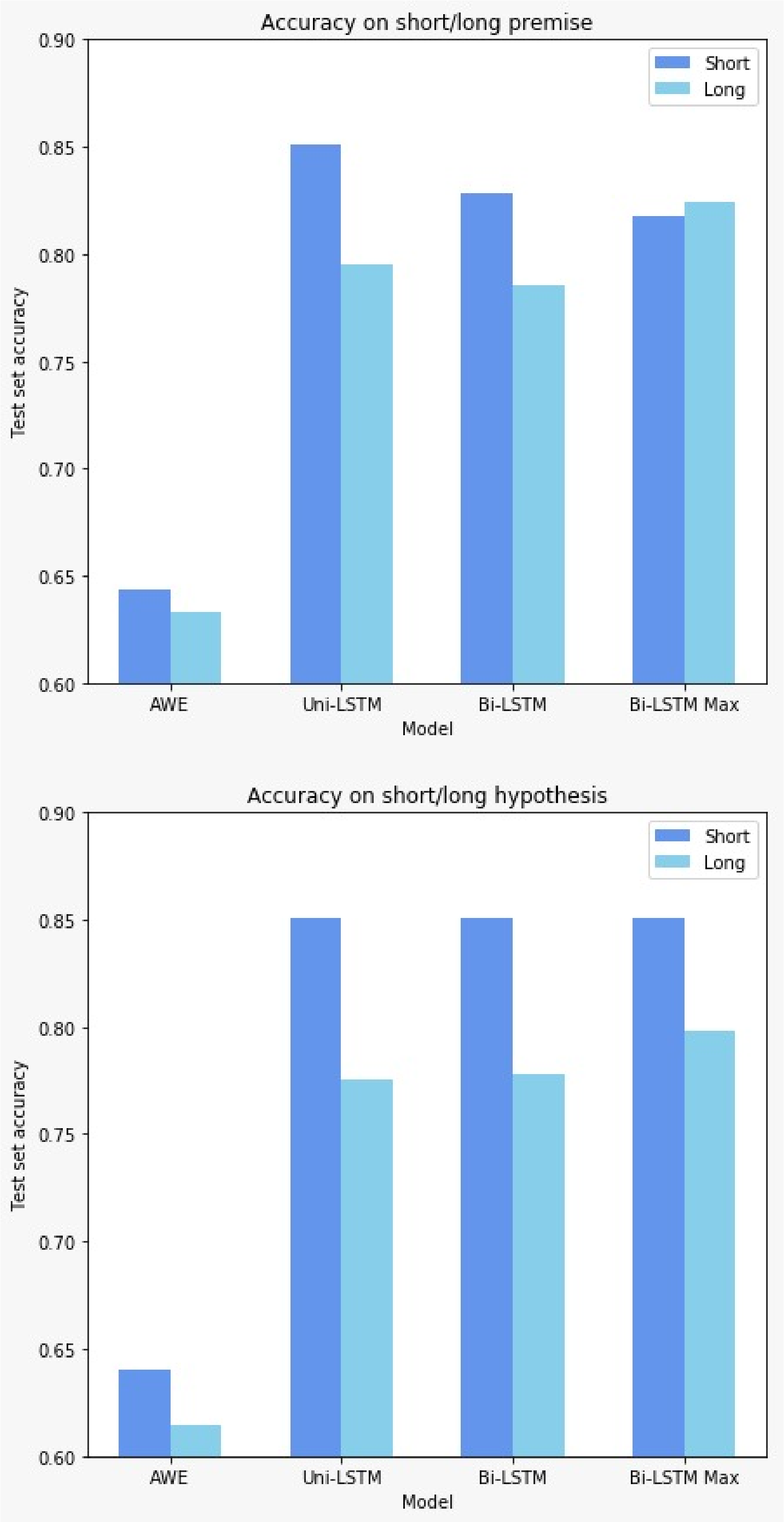
Accuracy on short/long premise
Short
Long
Test set accuracy
0.90
0.85
0.80
0.75
0.70
0.65
0.60
AWE
Uni-LSTM
Bi-LSTM
Bi-LSTM Max
Model
Accuracy on short/long hypothesis
Short
Long
Test set accuracy
0.90
0.85
0.80
0.75
0.70
0.65
0.60
AWE
Uni-LSTM
Bi-LSTM
Bi-LSTM Max
Model

Figure 1: