Natural Language Inference and Transfer Tasks

John Tsiamas (2597404) link to GitHub repository

MODEL	NLI				Transfer			
	dev loss	test loss	dev accu	test accu	dev macro	test macro	dev micro	test micro
BoW	0.8394	0.8398	0.6465	0.6463	-	-	-	-
uniLSTM	0.5114	0.5175	0.8029	0.7970	0.696	0.694	0.7034	0.7474
biLSTM	0.5119	0.5222	0.8007	0.7907	0.7186	71.3317	73.1308	76.8621
biLSTM + maxpooling	**0.4287**	**0.4358**	**0.8377**	**0.8318**	**0.7367**	**0.7321**	**0.7682**	**0.8079**

Table 1: Performance metrics on the Natural Language Inference (NLI) task and 6 other accuracy based classification tasks (MR, CR, SUBJ, TREC, MRPC, SICK-E)

MODEL	Training		Hyperparameters			
	epochs trained	best epoch	embeddings size	base learning rate	weight decay (classifier)	weight decay (encoder)
BoW	9	7	300	10^{-2}	10^{-2}	-
uniLSTM	9	6	2048	10^{-3}	10^{-2}	10^{-4}
biLSTM	6	3	4096	10^{-3}	$5 * 10^{-3}$	10^{-4}
biLSTM + maxpooling	19	16	4096	$5 * 10^{-4}$	10^{-3}	10^{-4}

Table 2: Training and Hyperparameters. On top of the encoders, a 1-hidden layer classifier was used with size 512 and a ReLU non-linearity. All models were trained using Adam.

MODEL	MR		CR		SUBJ		TREC		MRPC		SICK-E	
uniLSTM	0.707	0.6994	0.7721	0.7499	0.829	0.8225	0.3973	0.434	0.7267	0.72	0.744	**0.7386**
biLSTM	0.725	0.7191	0.7795	0.7671	0.8657	0.8641	0.4624	**0.486**	0.7311	0.7119	0.748	0.7317
biLSTM + maxpooling	0.805	**0.7893**	0.8524	**0.8376**	0.9057	**0.8985**	0.4259	0.434	0.7213	**0.7246**	0.71	0.7085

Table 3: Performance of encoders on transfer tasks. For the majority of the tasks (4/6) the biLSTM with max pooling achieves the best results. It seems that the models are not well tuned for retrieval since they achieve surprisingly low performance. Another highlight is the fact that the vanilla LSTM surpassed bot the bi-LSTM models in SICK-E, indicating that the bidirectional models outfitted the SNLI dataset.

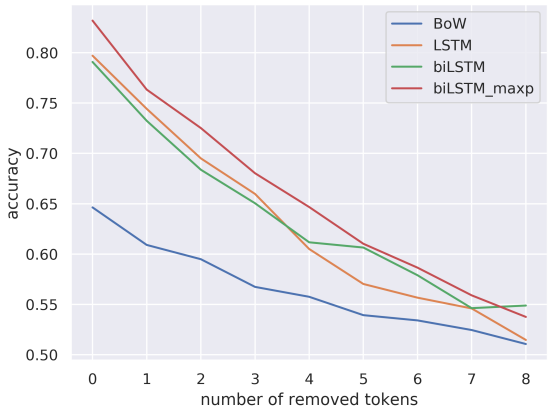

Figure 1: Accuracy of the 4 encoders on the test set, where in each sentence, x number of tokens have been randomly replaced with the unk token. We observe that all algorithms approach the Bag-of-words performance as there is more ambiguity in the sentence semantics.

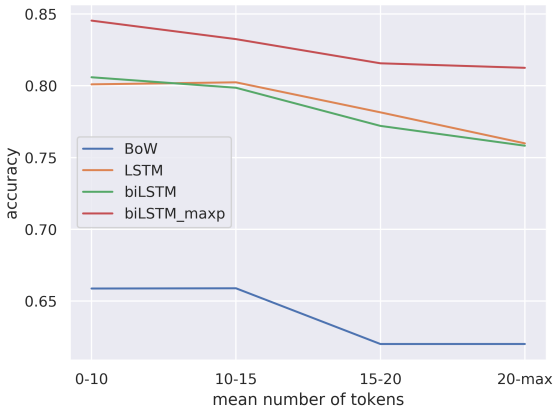

Figure 2: Accuracy of the 4 encoders on a test set based on the mean number of tokens in the sentence pair. As was hypothesized, performance is better on smaller sentences, where there is less ambiguity and better language comprehension.