Task-universal sentence embeddings from learning natural language inference

Yao Liu yaoliu@stanford.edu
Jeha Yang jeha@stanford.edu
Alex Kolchinski yakolch@stanford.edu
Katherine Yu yukather@alumni.stanford.edu

Abstract

In this project, we show how fixed-dimensional sentence embeddings from encoders trained on Stanford Natural Language Inference (SNLI) dataset (Bowman et al., 2015) and a new dataset we generated, derived from Stanford Question Answering (SQuAD) dataset (Rajpurkar et al., 2016) could be transfered into many other semantic tasks, especially tasks with little training data. The NLI baseline is based on Siamese from Conneau et al. (2017). Based on their learning and transfer evaluation framework, we use following models to learn our sentence embeddings and compare the results on both SNLI and transfer task: Siamese model (Conneau et al., 2017) with different kind of encoder: BiLSTM, Transformer (Vaswani et al., 2017), 2 layer BiLSTM, and Decomposable attention model (Parikh et al., 2016). We propose a NLI-like dataset derived from SQuAD to augment our training data and show benefit from multitask training with it in transfer task performance. Github repository: `https://github.com/kolchinski/NLI_2018`

1 Introduction

The Natural Language Inference task has been very well studied, most recently with the release of the SNLI dataset (Bowman et al., 2015), as well as the MultNLI dataset. Conneau et al. (2017) first showed that the natural language inference task was so universal that an encoder learned on NLI could produce fixed-dimensional sentence embeddings (e.g. through an elementwise-max or mean of the encoder hidden states) that could be used in transfer learning on many other semantic tasks, especially tasks with little data.

In this project, we follow the framework of learning sentence representation and transferring in Conneau et al. (2017) and explore potential improvement in several ways: We investigate some alternative choices of encoder model, especially different attention-related models, under the same training framework. We derive a new classification dataset from SQuAD dataset and using multitask training to achieve more task-universal representation.

2 Datasets

- **SNLI dataset**: Given a sentence pair (premise,hypothesis) we need to infer their relatedness, choosing from {entailment,contradiction,neutral}. The dataset has 549k/10k/10k balanced samples in train/dev/test set.
  (Ex) Two blond women are hugging one another.
  $+ \begin{cases} \text{There are women showing affection.} & \text{(entailment)} \\ \text{The women are sleeping.} & \text{(contradiction)} \\ \text{Some women are hugging on vacation.} & \text{(neutral)} \end{cases}$
- **"ClassifSQuAD"**: This is a new classification training dataset we derived from SQuAD Rajpurkar et al. (2016), inspired by NLI datasets. Given a tuple of sentences (question, answer1, answer2) we need to predict whether answer1 is correct. We generated positive samples from original QA pairs and negative samples from answers to other questions within the same article, checking that they are not the same or substrings.

Importantly, we filtered any examples which had OOV's in either the question or answer and any answer with fewer than 5 words since shorter answers were mostly undescriptive named entities. The dataset totally has 745k/128k samples in train/dev set, and 13.2k/2.1k unique questions in train/dev set.

(Ex) What changes macroscopic closed system energies?

$$+ \begin{cases} \textbf{internal energies of the system}(correct) \\ \text{directed toward the center of the curving path (wrong)} \end{cases}$$

(Ex) For what cause is money raised at the Bengal Bouts tournament at Notre Dame?

$$+ \begin{cases} \textbf{the holy cross missions in bangladesh}(correct) \\ \text{a golden statue of the virgin mary (wrong)} \end{cases}$$

(Ex) What was the cost for a half minute ad? $+ \begin{cases} \textbf{\$ 5 million for a 30-second}(correct) \\ \text{newton was limited by denver 's defense (wrong)} \end{cases}$

3 Methods

The goal of this project is to learn general sentence encoder through supervised tasks including SNLI and ClassifiSQuAD, a classification tasks derived from SQuAD. These supervised tasks take multiple sentences as input and need to predict a 2/3-class label. Our model trained in supervised learning tasks could be roughly divide into two parts: encoding parts which process each sentence separately but share the same parameters, and classification part which jointly uses embeddings of all sentences to predict the label. We take the encoding part as a task-universal encoder and test its output embeddings in transfer task later.

In this section, we describe the model structure that we trained in SNLI task and ClassifiSQuAD. In the next section we will describe more details about the combined training process. We use the Siamese framework which learns a shared encoder for premise and hypothesis sentences, followed by a MLP classifier. We investigated different choices of encoder: bidirectional LSTM, 2 Layer bidirectional LSTM, Transformer attention model. Besides Siamese framework we also investigated a more attention based method: decomposable attention model Parikh et al. (2016).

For all of those models, we use GloVe 840B.300D word embeddings Pennington et al. (2014) [1] and froze it during training. Because this task relies so heavily on pretrained word embeddings, we did not try subword/BPE methods. We also tried different kinds of aggregation (elementwise-mean and -max), to aggregate encoder hidden states over timesteps into a fixed-dimensional sentence embedding Conneau et al. (2017).

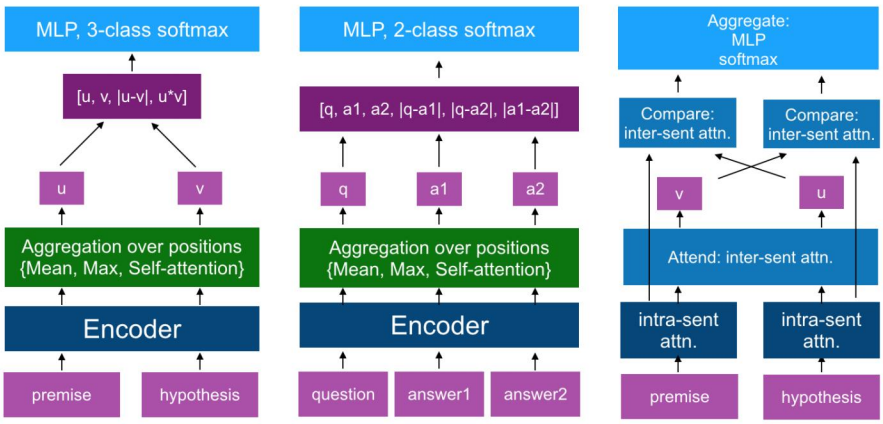

Figure 1: NLI Siamese architecture (left), Classif SQuAD Siamese architecture (mid) and Decomposable attention architecture

[1] https://nlp.stanford.edu/projects/glove/

3.1 Siamese Framework

Siamese models are by far not the best-performing models on the two training tasks due to not using intersentence word-by-word attention; however, we need to use siamese training for to produce generic sentence embeddings since sharing the same encoder parameters allows the single encoder to learn from all sentences in the training data and at inference on a single sentence, we do not have a target sentence (we cannot use seq2seq attention).

Siamese separately learns an encoder of the individual sentences, sharing the same encoder parameters. After aggregating over positions in one sentence, we get a fix dimensional representation of sentence: u and v. We extract relations between u and v : (i) concatenation of the two representations (u, v); (ii) element-wise product $u * v$; and (iii) absolute element-wise difference $|uv|$, then use a feed-forward network as classifier over these features. Figure 1 shows the training framework of Siamese model in both NLI dataset and SQuAD dataset.

One branch of our work focuses on investigating different kinds of encoders in Siamese framework. We tried the following encoder architectures with Siamese training:

- Bidirectional LSTM (Conneau et al., 2017). Bidirectional LSTM output the concatenation of a forward LSTM and a backward LSTM that read the sentences in two opposite directions. Conneau et al. (2017) show that better than LSTM, GRU and Hierarchical ConvNet encoders in terms of learning transferable representation.
- 2-layer Bidirectional LSTM. We tried to stack 2 BiLSTM in order to extract higher level feature.
- Transformer Vaswani et al. (2017). We modified the original transformer Vaswani et al. (2017) so that the encoders for two sentences have the same structure and share parameters.

3.2 Decomposable Attention

Model with more inter-sentence word-by-word attention could achieve better result on the original supervised tasks. So we also investigate decomposable attention model (Parikh et al., 2016). Decomposable attention model first do intra-sentence attention to get the sentence embeddings. Second, it soft-align the elements of two sentences using a variant of neural attention. The it compare the aligned subphrase with original sentence embeddings, using a feed-forward network. The last step is aggregate the output and feed the result through a final feed-forward network classifier. See Figure 1 for the structure. To obtain the sentence embeddings for the transfer task, we take the intra-sent attention part as encoder and froze it to output sentence embeddings.

4 Multitask with a new dataset derived from SQuAD

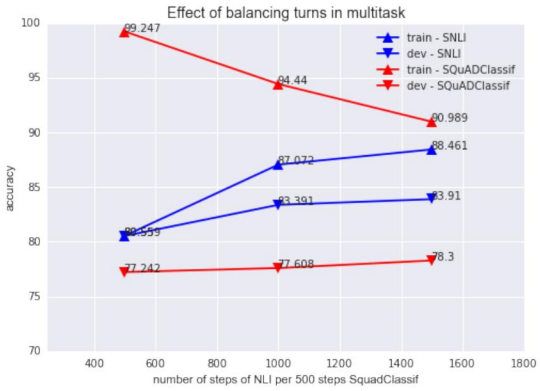

Figure 2: Multitask training - effect of balancing training steps between tasks

We (unconventially) train each of the two tasks for many consecutive steps (e.g. 1000 steps NLI, 500 steps Classif-SQuAD) and find that each task can recover and improve on its previous train loss very quickly (within 100 batches)

after the other tasks's turn (the reason for this initially was that we wanted to compare in-epoch progress against a reference single-task learning curve). It seemed actually important to use different batch sizes for the two tasks: 64 for NLI and 128 for ClassifSQuAD; thus, we generally take fewer steps in ClassifSQuAD. Figure 2 shows how the number of steps we train on SNLI per 500 SQuAD steps.

Table 1: SNLI and ClassifSQuAD Results

Model	Trainable params	SNLI			ClassifSQuAD	
		Train acc	Dev acc	Test acc	Train acc	Dev acc
BLSTM (2048 hidden size)	47M	84.122	83.350	83.429	-	-
2-BLSTM (1024 hidden size)	40M	85.556	83.062	82.504	-	-
Multitask LSTM (2048 hidden size)	32M	87.072	83.393	82.874	90.989	78.3
Multitask BLSTM (2048 hidden size)	62M	86.948	84.062	83.326	97.841	**80.717**
Decomposable Attention [2]	580K	83.062	**84.088**	**83.926**	-	-
Siamese Transformer (4-layer) [3]	989K	83.565	82.692	82.597	-	-

5 Task-universal sentence embedding

Conneau et al. (2017) showed that an encoder trained on NLI could be used to produce a task-universal sentence embedding, typically through the embedding produced by the element-wise maximum of the encoder output states of the sentence (max-pooling). To evaluate the quality of sentence representation we learned from SNLI/SQuAD, we use them as features in the 9 test tasks that are used as benchmark in Conneau et al. (2017)'s work:

- Sentence classification: sentiment analysis (MR, SST), product reviews (CR), subjectivity/objectivity (SUBJ) and opinion polarity (MPQA).
- Semantic inference: SICK-E(Entailment), SICK-R(Relatedness).
- Semantic textual similarity: STS14.

We use the sentence evaluation tools [4] from Conneau et al. (2017), which fit a simple MLP over learned features. We also compare the results of aggregating encoder output into a fixed length embeddings using max-pooling, as suggested in Conneau et al. (2017), and concatenation of mean and max-pooling.

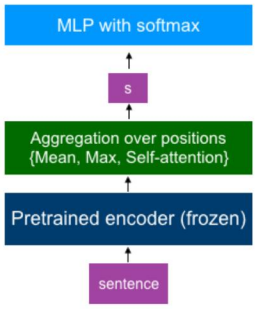

Figure 3: Transfer learning architecture

5.1 Self-attention

In addition to the MLP we tried to train task-specific self-attention which takes the place of mean- and max- aggregations by attending pairwise word-to-word within the same sentence and aggregating to a fixed-dimensional sentence

[2] Our code uses code from https://github.com/libowen2121/SNLI-decomposable-attention

[3] Our code uses code from https://github.com/jadore801120/attention-is-all-you-need-pytorch

[4] https://github.com/facebookresearch/SentEval

embedding Lin et al. (2017), which goes to the MLP. This did not do very well—we didn't have enough time to tune but because the hidden size are large, the many extra parameters (W_1 scales with hidden size) cause overfitting. Also the increased size of the sentence embedding causes the MLP to overfit. The architecture is, for parameters $W_1 \in \mathbb{R}^{d_a \times (2*)hiddensize}$, $W_2 \in \mathbb{R}^{r \times d_a}$, where d_a, r are hyperparameters (we tried $d_a = 128, r = 3$)

$$A = softmax(W_2 tanh(W_1 H^T)),$$

where AH is the fixed dimensional sentence embedding of (flattened) size $(2*)hiddensize * r$, where $(2*)$ is for bidirectional.

Table 2: Transfer Tasks Results

Model	k [5]	MR	CR	MPQA	SUBJ	SST-B	SST-F	SICK-E	SICK-R	STS14
Conneau et al. (2017) BLSTM(max)	4096	79.9	84.6	**89.8**	92.1	**83.3**	-	**86.3**	**0.885**	.68/.65
Pagliardini et al. (2017) Sent2Vec	700	75.8	80.3	85.9	91.2	-	-	-	-	.65/**.67**
BLSTM(max)	2048	81.32	84.11	89.19	93.02	81.0	42.08	85.18	0.8727	.68/.65
BLSTM(max,mean)	2048	81.33	84.56	89.3	92.19	80.45	40.23	85.83	0.8812	.66/.64
2-BLSTM(max)	1024	80.71	83.21	88.86	91.86	74.9	37.1	83.93	0.868	.68/.64
2-BLSTM(max,mean)	1024	81.2	83.79	89.02	92.49	76.28	39.28	84.96	0.875	.66/.64
Transformer(max)	512	72.57	76.06	87.12	89.3	75.23	39.68	82.59	0.8467	.66/.64
Transformer(max,mean)	512	74.22	76.24	87.85	90.6	76.99	41.4	82.79	0.8557	.63/.62
Multitask LSTM(max)	2048	80.9	**84.82**	89.68	92.74	80.23	42.44	85.41	0.8695	**.69/.67**
Multitask BLSTM(max)	2048	**81.82**	84.21	89.38	**93.77**	81.0	**42.58**	85.53	0.8740	.68/.65
Multitask LSTM(selfatten)	6144	76.89	78.26	87.71	91.87	-	-	-	-	-
Decomposable Att (Max)	200	70.59	74.89	86.63	87.38	72.87	35.79	79.26	0.817	.41/.44
Decomposable Att (Sum)	200	73.52	76.95	86.27	89.66	78.36	39.05	74.1	0.767	.56/.55
Decomposable Att (Max,Mean)	200	73.4	77.01	87.84	89.75	76.33	39.14	80.6	0.825	.60/.58

6 Conclusion and Future Work

- Multitask training seems promising.
- Siamese methods with BiLSTM encoder from Conneau et al. (2017) achieved best performance in some tasks, while Multitask training derived from SQUAD dataset did so in other tasks.
- Concatenation of max-pool and mean-pool could help many encoders achieve better transfer performance.
- Decomposable attention model does well on NLI dataset but could not learn good transferable embeddings because they rely on inter-sentence attention.

For future work, we have a lot of work to do on multitask training, such as alternating turns conventionally by sampling so each task doesn't have to waste as much time "catching up" to the encoder chagnes. We would try adding additional training data such as MultNLI, Quora Question Pairs.

The fixed GloVe word embeddings are very important to this model. We can try other word embeddings. We might theoretically want to reach a point where we have enough tasks and data to train word embeddings through backpropagation.

We also want to work on a decomposition analysis on the set of transfer/SentEval tasks so we can understand performance by important task characteristics like sentence length, OOV rate, and relative size of the training data.

Most importantly, we want to add a bottle layer to the encoder training so that we can compare transfer learning based on the same sentence embeddding size. We have two confounding factors of size of the encoder (hidden size) versus overfitting to large sentence embeddings, and we want to separate these to derive insights.

[5] size of sentence embeddings

References

S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*, 2015.

A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes. Supervised learning of universal sentence representations from natural language inference data. *arXiv preprint arXiv:1705.02364*, 2017.

Z. Lin, M. Feng, C. N. dos Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio. A structured self-attentive sentence embedding. *eprint arXiv:1703.03130*, 2017.

M. Pagliardini, P. Gupta, and M. Jaggi. Unsupervised learning of sentence embeddings using compositional n-gram features. *eprint arXiv:1703.02507*, 2017.

A. P. Parikh, O. Täckström, D. Das, and J. Uszkoreit. A decomposable attention model for natural language inference. *arXiv preprint arXiv:1606.01933*, 2016.

J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014. URL `http://www.aclweb.org/anthology/D14-1162`.

P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. pages 6000–6010, 2017.