Natural Language Inference Task

Using Pre-trained Embedding Layers

Mrinal Rawool, Matthew Penne, Mason Lien, Ketemwabi Yves Shamavu

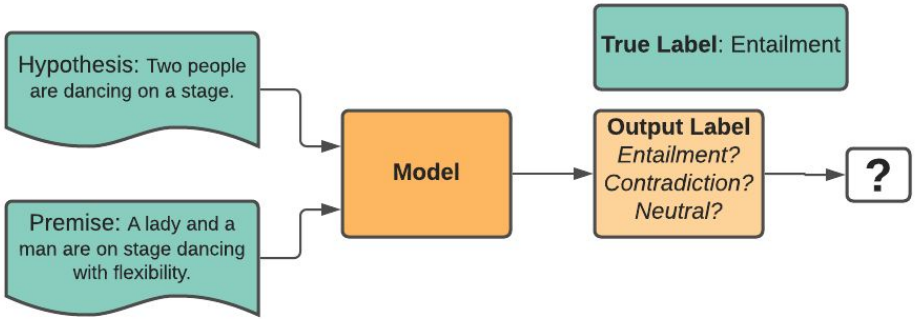
Contents

Problem Description and Scope

The Task: Given a premise, can a hypothesis be justifiably inferred from it?
Our Goal: Design a fast and accurate model to predict semantic relationship between a pair of sentences.

Scope: General text written in English.

- The corpus used for this project is Stanford Natural Language Inference (SNLI) dataset.
- Hosted by Tensorflow Datasets
- The language in the dataset is English as spoken by users of the website Flickr and as spoken by crowdworkers from Amazon Mechanical Turk



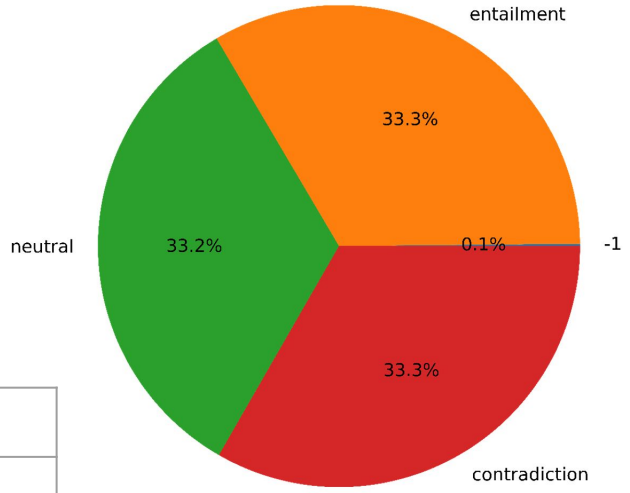
Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. “A large annotated corpus for learning natural language inference.” 2015.

Data set description

- The dataset consist of three splits: train, test, and validation.
- Each element in the dataset contains
  - A Premise (tf.string),
  - A hypothesis (tf.string)
  - A label (tf.int64)
- The sentence pairs belong to one of three classes
  - Label 0 = entailment
  - Label 1 = neutral
  - Label 2 = contradiction
- Each split is balanced
- ~500K train, 10K test, 10K validation sentence pairs

Data Cleaning and Preprocessing

- General preprocessing
  - Presence of invalid labels (-1)
  - Sentence preprocessing handled by tokenizer
- Preprocessing for BERT
  - Needs inputs in a specific format
  - Max length 512, BERT embedding dim = 768
  - Need to limit the length of sequences (currently set to 128)



Input	This is sentence A	This is sentence B
Input Word IDs	[101, 22, 45, 400, 2, 102]	[22, 45, 400, 3, 102]
Input Type IDs	[0,0,0,0,0,0]	[1,1,1,1,1]
Attention Mask	[1,1,1,1,1,1]	[1,1,1,1,1]

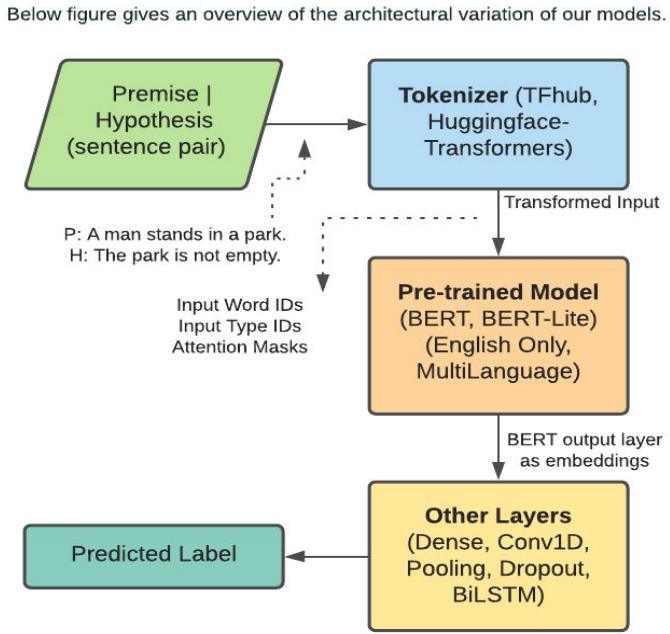
Model Architecture

Why BERT?

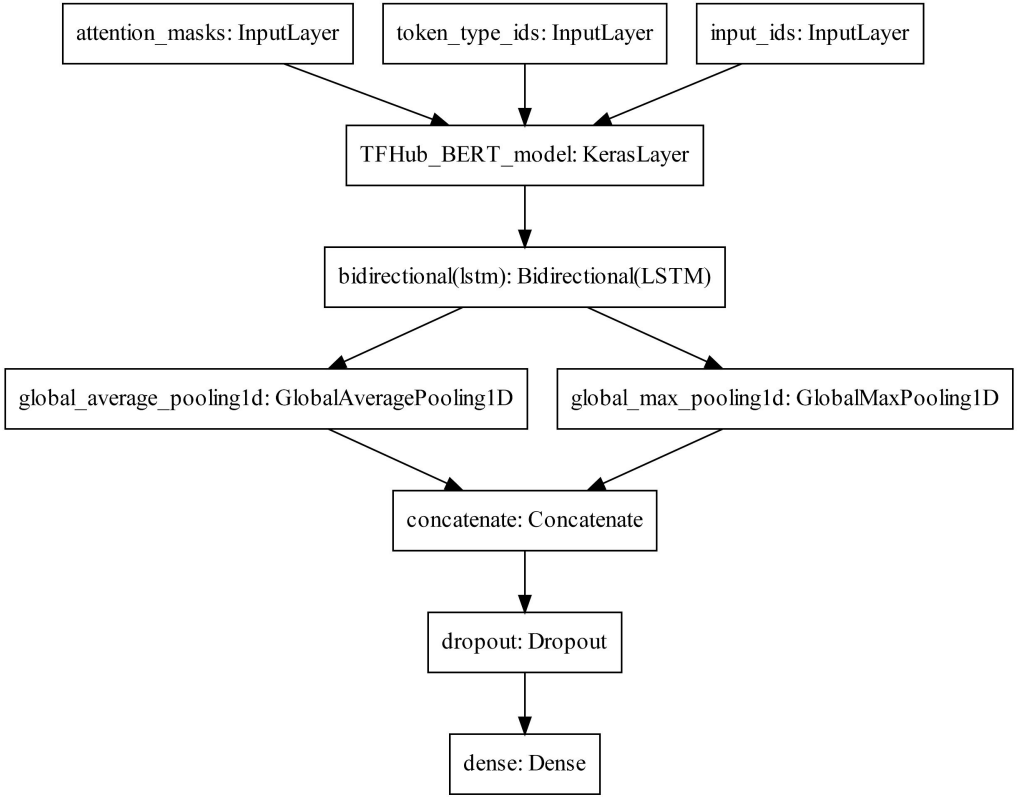
- Trained on MLM and NSP tasks
- Context awareness necessary
- Preferred choice for NLU
- Offers a tokenizer and a pretrained model

Tokenizer:

- Vocabulary file
- Preprocessing options such as case conversion, adding tokens
- Allows basic and ‘wordpiece’ tokenization.
  - E.g This is sentence A
  - [‘this’,’is’,’sentence’,’a’] (Basic)
  - [‘this’,’is’,’sen’,’#ten’,’#ce’,’a’] (wordpiece)



BERT



Baseline BERT Model

BERT Variations

- Tiny BERT
  - Trained on the English slice of Wikipedia and BooksCorpus
  - Smallest version of BERT

- AlBERT (A Lite BERT)
  - It has less parameters allowing for large architectures without greatly increasing the training time
  - It also achieves better behavior with respect to model degradation

- MNLI BERT (Multi-Genre NLI)
  - Since it was trained on more formal text, it does not do well with colloquial text such as social media or messages
  - Performs well on NLI tasks

Tiny-BERT

L=2: the number of layers

H=128: the hidden size

A=2: the self-attention head

```
Total params: 4,485,508
Trainable params: 99,587
Non-trainable params: 4,385,921
```

	H=128	H=256	H=512	H=768
L=2	**2/128 (BERT-Tiny)**	2/256	2/512	2/768
L=4	4/128	**4/256 (BERT-Mini)**	**4/512 (BERT-Small)**	4/768
L=6	6/128	6/256	6/512	6/768
L=8	8/128	8/256	**8/512 (BERT-Medium)**	8/768
L=10	10/128	10/256	10/512	10/768
L=12	12/128	12/256	12/512	**12/768 (BERT-Base)**

Turc, Iulia, et al. “Well-Read Students Learn Better: On the Importance of Pre-Training Compact Models.” Sept. 2019.

AlBERT

L=24

H=1024

A=16

```
Total params: 17,946,627
Trainable params: 262,659
Non-trainable params: 17,683,968
```

Changes from BERT

- Factorized embedding parameterization
  - Reduce large vocabulary matrix into 2 smaller ones
- Cross layer parameter sharing
- Uses sentence order prediction (SOP) loss instead of next sentence prediction (NSP) loss

Lan, Zhenzhong, et al. “ALBERT: A Lite BERT for Self-Supervised Learning of Language Representations.” Feb, 2020

MNLI-BERT

L=12

H=768

A=12

```
Total params: 109,909,508
Trainable params: 427,267
Non-trainable params: 109,482,241
```

Initially trained on the Wikipedia and BooksCorpus.

Fine tuned on the MNLI data set for 12 epochs

Williams, Adina, et al. “A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference.” 2018.

Devlin, Jacob, et al. “BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding.” May, 2019.

Other Layers

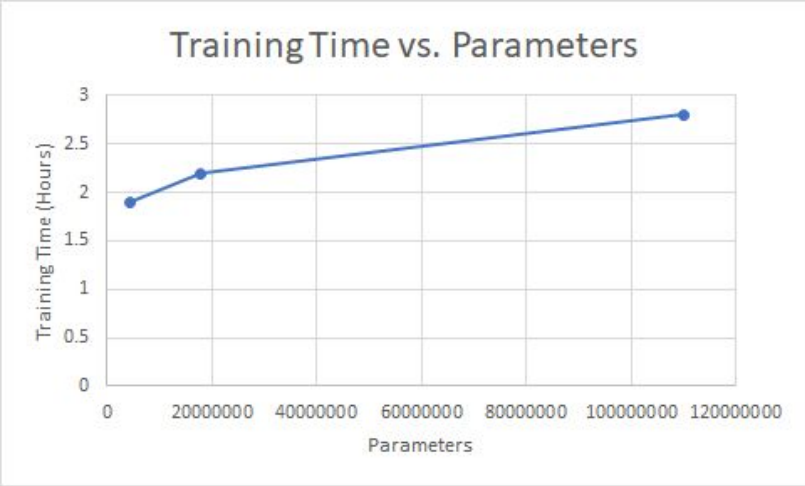
- Bidirectional Long Short-Term Memory (LSTM)
  - 64 units
- Global Average and Global MaxPool layers
- Concatenate layer
- Dense Layer
  - Softmax activation
  - Output of 3 labels
  - Entailment (0), neutral (1), contradiction (2)

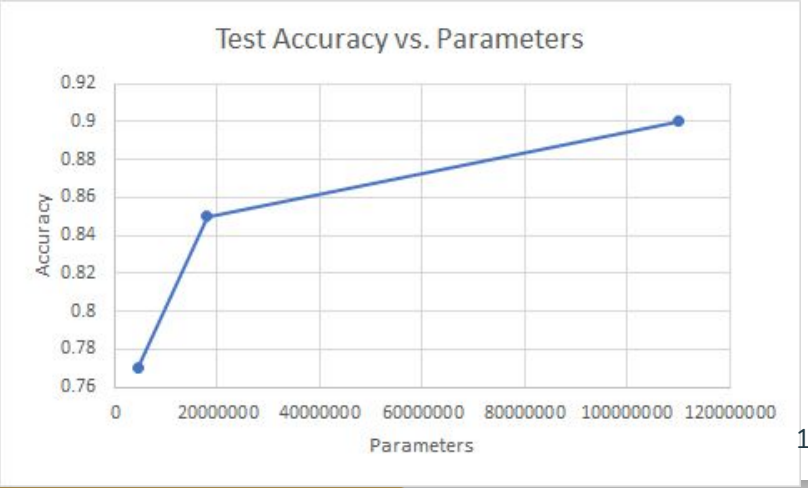
Hyperparameters

- Batch size=32
- Adam optimization with learning rate 1e-5
- Train 2 epochs without training BERT
- Train 2 epochs with BERT training (fine tuning)

Results

BERT	Total Training Time (hours)	Test Accuracy after 4 epochs
Tiny-BERT	1.9	77%
AlBERT	2.2	85%
MNLI-BERT	2.8	90%



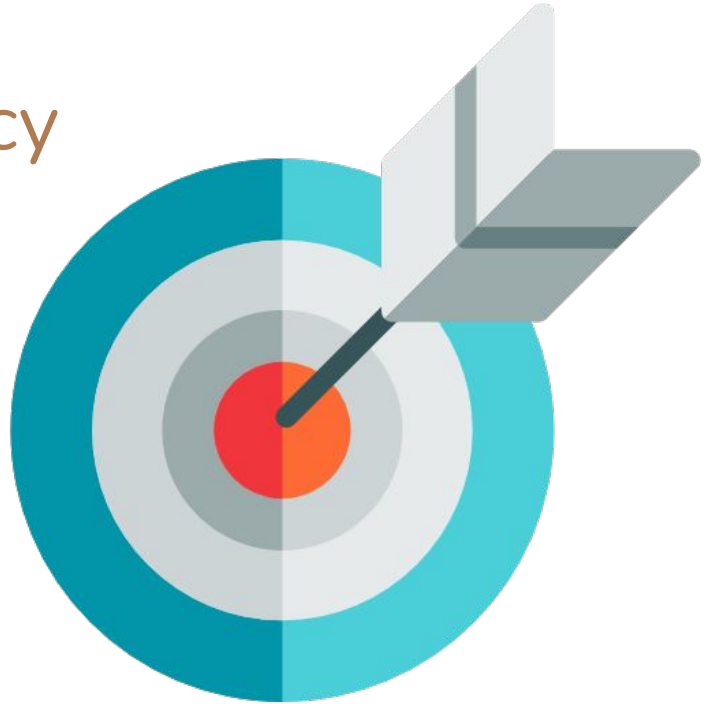


Increase Training Speed

- More fine tuning on small BERT variants
  - RoBERTa: A robustly optimized BERT pre-training approach and lite BERT
- Don’t use all training data
  - Assess the performance on a fraction of the 550k training samples
- Adjust the max sentence length
  - Small => loss of context
  - Large => huge embedding dimension

Increase Training Accuracy

- Fine Tuning
- Hyper-parameters:
  - Epochs
  - Sequence length
- Using grid search / random search
- Additional Layers
  - Convolutional Layers
  - Regularization
  - Pooling
  - GRU



Challenges

- Issues with APIs of pretrained BERT from Hugging Face: Threw several errors which led us to stick with TFHub models
- Pickle/np.save: Tried to save intermediate results as picke files to allow load on demand. Caused OOM issues due to the huge embedding dim (768) as well as the dataset size
- Tensorflow Text installation issues resulting in reinstalling TF and messing up with the environment

Future Work

- Grid/randomized search to find out how many additional layers are optimal to add after the BERT pre-trained model
- Try out different preprocessing steps and build custom embedding layers
- Compare the performance of more BERT models from TFHub on the this NLI task

Thank you!

Any Questions?