Siamese BERT-Networks

Richer Sentence Embeddings using Sentence-BERT — Part I

Using naive sentence embeddings from BERT or other transformer models leads to underperformance. To get around this, we can fine-tune BERT in a siamese fashion. The result is a rapid generation of rich sentence embeddings.

B benchmarks [2].

In many cases, it outperformed human performance [3].

Fast-forward 1 year along, and several improved variants of BERT [4][5][6][7][8] have popped up, with new ones being released by large tech companies seemingly every month.

We will first briefly review BERT (a more in-depth review is here), and then explain how to efficiently generate rich sentence embeddings using BERT.

Specifically, we will discuss a recent paper from UKP (Ubiquitous Knowledge Processing Lab): Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks [9]

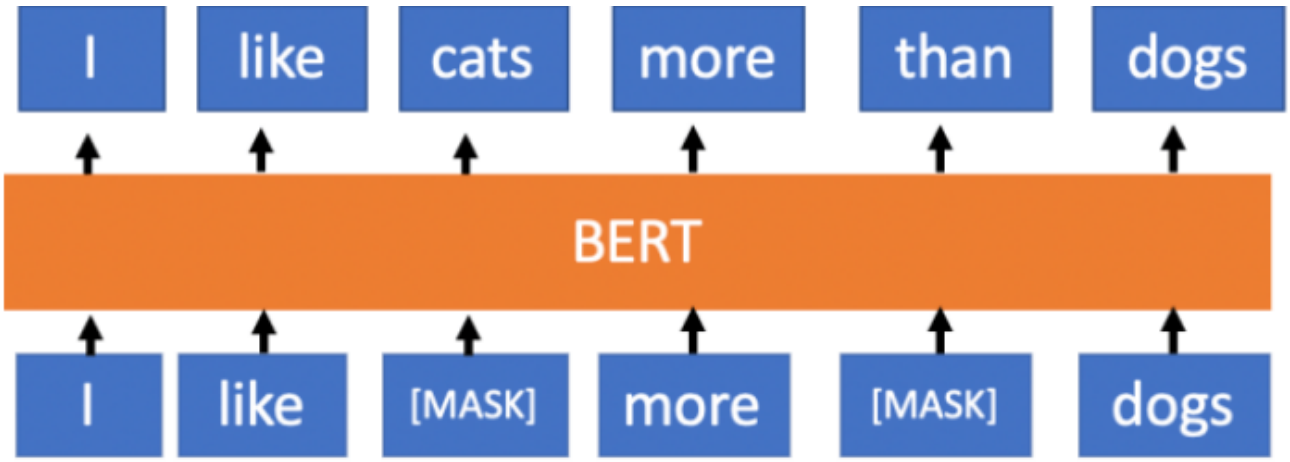
In part II of this post, we will implement an MVP of this strategy in PyTorch.

. . .

1. Preliminaries: BERT is trained to give rich word embeddings

BERT is very good at generating word embeddings (word vectors) that are rich in semantics and depend heavily on context.

The sentences “I ate an apple” and “Apple acquired a startup” will have completely different word embeddings for “apple” generated by BERT, due to the context of the words.



BERT reconstructs partially masked sentences using the context of the surrounding sentence (Masked language modeling)

The original BERT also uses a next-sentence prediction objective, but it was shown in the RoBERTa paper [8] that this training objective doesn't help that much.

In this way, BERT is trained on gigabytes of data from various sources (e.g much of Wikipedia) in an unsupervised fashion.

. . .

2. (Old) Sentence Embedding Methods are not Rich

For many NLP tasks, we need sentence embeddings. This includes, but is not limited to, semantic similarity comparison, sentence clustering within documents and information retrieval via semantic search.

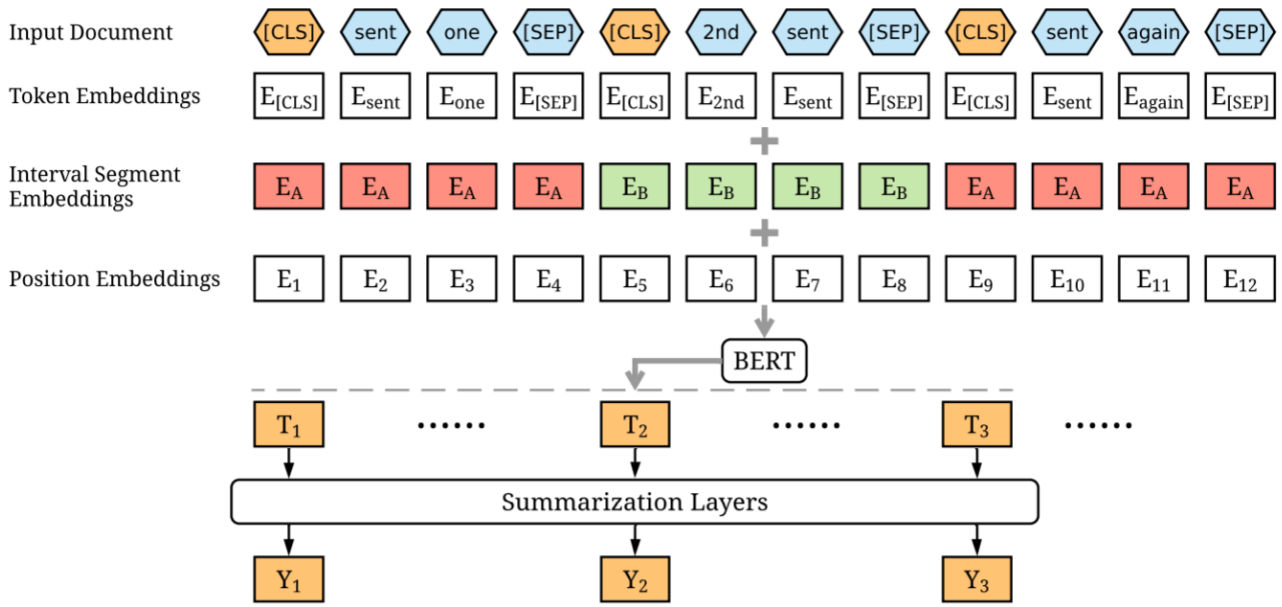
At Genei, we make use of sentence embeddings to cluster sentences in documents, which

$$s = \frac{1}{|S|} \sum_{w \in S} w$$

We average the word embeddings in a sentence to get the sentence embedding

2b. Using the [CLS] vector as the sentence embedding

Alternatively, we can use the embedding for the [CLS] special token that appears at the start of the sentence.



The [CLS] token (shown in orange) is used as a sentence embedding in this paper [12] that uses BERT for extractive summarization

Model	STS12	STS13	STS14	STS15	STS16	STSb	SICK-R	Avg.
Avg. GloVe embeddings	55.14	70.66	59.73	68.25	63.66	58.02	53.76	61.32
Avg. BERT embeddings	38.78	57.98	57.98	63.15	61.06	46.35	58.40	54.81
BERT CLS-vector	20.16	30.01	20.09	36.88	38.08	16.50	42.63	29.19
InferSent - Glove	52.86	66.75	62.15	72.77	66.87	68.03	65.65	65.01
Universal Sentence Encoder	64.49	67.80	64.61	76.83	73.18	74.92	**76.69**	71.22
SBERT-NLI-base	70.97	76.53	73.19	79.09	74.30	77.03	72.91	74.89
SBERT-NLI-large	72.27	**78.46**	**74.90**	80.99	76.25	**79.23**	73.75	76.55
SRoBERTa-NLI-base	71.54	72.49	70.80	78.74	73.69	77.77	74.46	74.21
SRoBERTa-NLI-large	**74.53**	77.00	73.18	**81.85**	**76.82**	79.10	74.29	**76.68**

Table 1: Spearman rank correlation ρ between the cosine similarity of sentence representations and the gold labels for various Textual Similarity (STS) tasks. Performance is reported by convention as $\rho \times 100$. STS12-STS16: SemEval 2012-2016, STSb: STSbenchmark, SICK-R: SICK relatedness dataset.

Results from the Sentence-BERT paper [9]

. . .

3. SentenceBERT: Fine-tuning BERT to give good Sentence Embeddings

The idea is to fine-tune BERT sentence embeddings on a dataset which rewards models that generates sentence embeddings that have the following property:

> When the cosine similarity of the pair of sentence embeddings is computed, we want it to represent accurately the semantic similarity of the two sentences.

3a. (Pre)-Training Strategy: Siamese Neural Network

The general idea introduced in [9] is to pass 2 sentences through BERT, in a siamese fashion. A good diagrammatic summary is below:

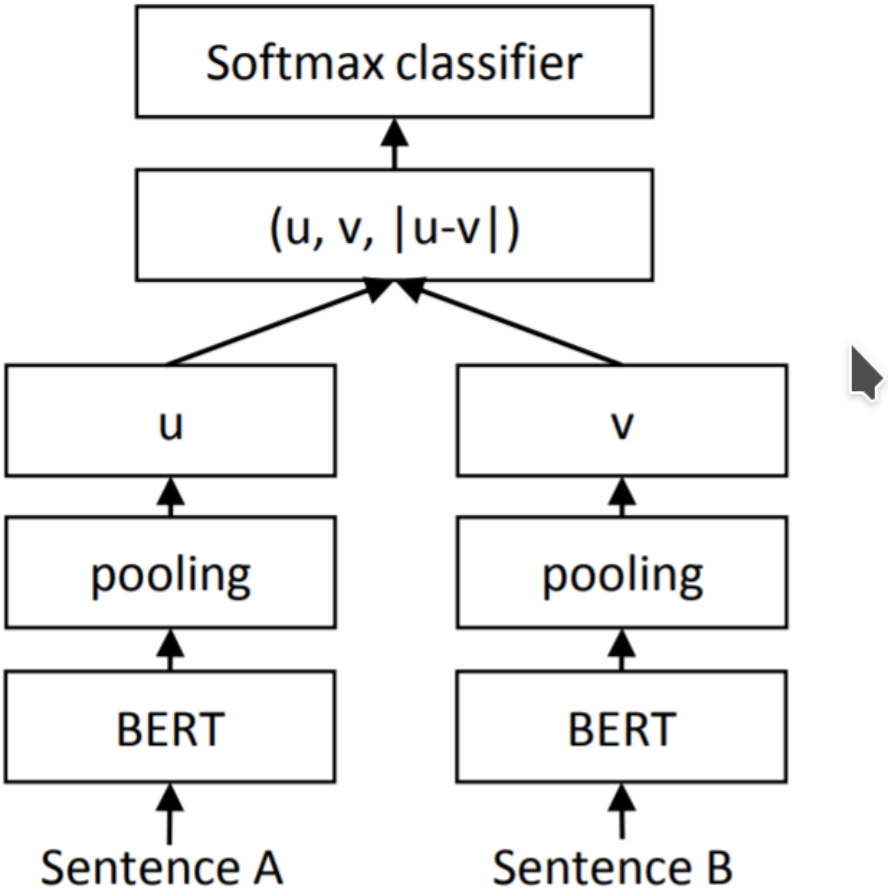


Figure 1: SBERT architecture with classification objective function, e.g., for fine-tuning on SNLI dataset.

trainable weight matrix $W \in \mathbb{R}^{3N \times K}$, where N is the sentence embedding dimension, and K is the number of labels.

3b. Ablation study for pooling and concatenation strategies

The **pooling** operation is flexible, although the researchers found that a *mean aggregation* worked best (compared to a max or CLS aggregation strategy).

Several **concatenation strategies** were tried as well; (u, v, ||u-v||) worked the best.

Ablation results from the paper are shown below:

	NLI	STSb
Pooling Strategy		
MEAN	**80.78**	**87.44**
MAX	79.07	69.92
CLS	79.80	86.62
Concatenation		
(u, v)	66.04	-
$(\|u - v\|)$	69.78	-
$(u * v)$	70.54	-
$(\|u - v\|, u * v)$	78.37	-

At inference, we compute sentence embeddings and then compute the cosine similarity of the respective pairs of sentences we want to compute the semantic textual similarity of:

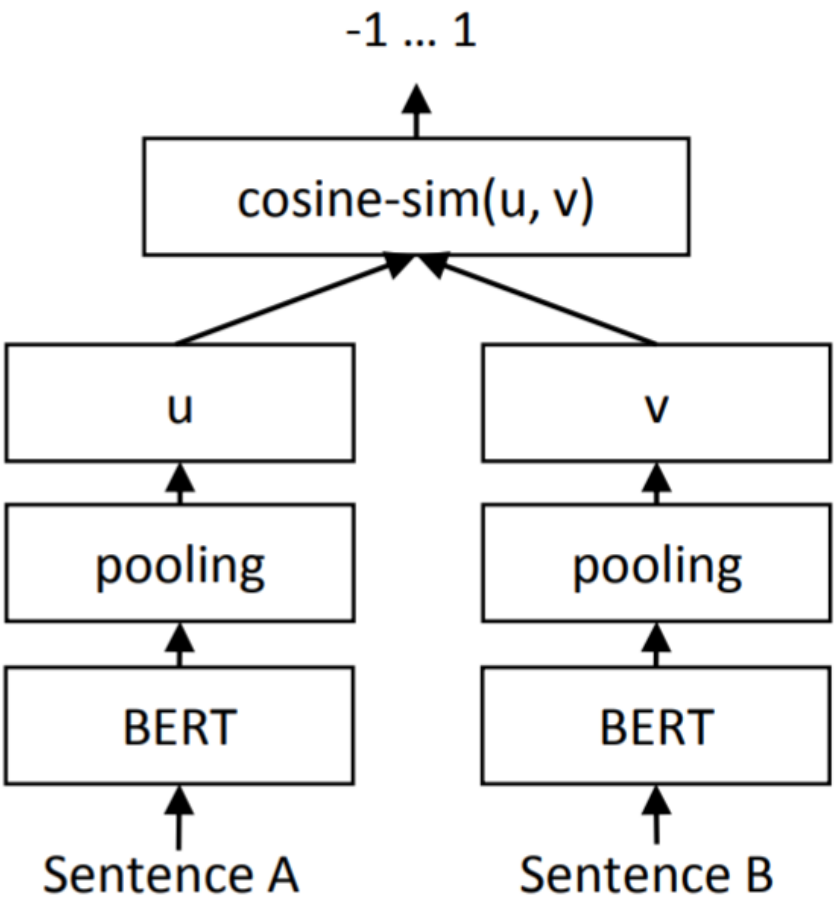


Figure 2: SBERT architecture at inference, for example, to compute similarity scores. This architecture is also used with the regression objective function

The output of the siamese network was trained to match that of a group of labeled datasets: the **STS benchmarks** [13]. These datasets provide labels from 0 to 5 for the semantic relatedness of a pair of sentences:

5	*The two sentences are completely equivalent, as they mean the same thing.*
	The bird is bathing in the sink. Birdie is washing itself in the water basin.
4	*The two sentences are mostly equivalent, but some unimportant details differ.*
	Two boys on a couch are playing video games. Two boys are playing a video game.
3	*The two sentences are roughly equivalent, but some important information differs/missing.*
	John said he is considered a witness but not a suspect. "He is not a suspect anymore." John said.
2	*The two sentences are not equivalent, but share some details.*
	They flew out of the nest in groups. They flew into the nest together.
1	*The two sentences are not equivalent, but are on the same topic.*
	The woman is playing the violin.

The SNLI (Stanford Natural Language Inference) dataset contains 570k human-written English sentence pairs manually labeled (by Amazon Mechanical Turk Workers) for balanced classification with the labels: *entailment, contradiction, neutral.*

Here are a few example pairs taken from the development portion of the corpus. Each has the judgments of five mechanical turk workers and a consensus judgment.

Text	Judgments	Hypothesis
A man inspects the uniform of a figure in some East Asian country.	contradiction C C C C C	The man is sleeping
An older and younger man smiling.	neutral N N E N N	Two men are smiling and laughing at the cats playing on the floor.
A black race car starts up in front of a crowd of people.	contradiction C C C C C	A man is driving down a lonely road.
A soccer game with multiple males playing.	entailment E E E E E	Some men are playing a sport.
A smiling costumed woman is holding an umbrella.	neutral N N E C N	A happy woman in a fairy costume holds an umbrella.

Training examples from SNLI dataset

3e. Final Results of the Paper

We'll quickly take a look at the final results the paper obtains:

Model	Spearman
Not trained for STS	
Avg. GloVe embeddings	58.02
Avg. BERT embeddings	46.35
InferSent - GloVe	68.03
Universal Sentence Encoder	74.92
SBERT-NLI-base	77.03

Trained on NLI data + STS benchmark data	
BERT-NLI-STSb-base	**88.33** ± 0.19
SBERT-NLI-STSb-base	85.35 ± 0.17
SRoBERTa-NLI-STSb-base	84.79 ± 0.38
BERT-NLI-STSb-large	**88.77** ± 0.46
SBERT-NLI-STSb-large	86.10 ± 0.13
SRoBERTa-NLI-STSb-large	86.15 ± 0.35

Table 2: Evaluation on the STS benchmark test set. BERT systems were trained with 10 random seeds and 4 epochs. SBERT was fine-tuned on the STSb dataset, SBERT-NLI was pretrained on the NLI datasets, then fine-tuned on the STSb dataset.

Final results...

Clearly, fine-tuning on both NLI + STS results in the best models. Interestingly enough, using RoBERTa [8] doesn't seem to help that much over BERT…

Finally, note the improvement we get over using the average BERT embeddings (line 2 of the table). The result is a step improvement.

. . .

4. Looking Forward — Part II

Instagram: here

Email: info@genei.io

Genei is an Ed-tech startup working on improving the productivity of students and academics by harnessing the power of NLP.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In NAACL-HLT

[2] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, Samuel R. Bowman. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding

[3] John Pavlus. Machines Beat Humans on a Reading Test. But Do They Understand? Quanta Magazine

[4] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A lite BERT for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942, 2019.

[5] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. arXiv e-prints.

[6] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and

optimized bert pretraining approach. ArXiv, abs/1907.11692, 2019.

[9] Reimers, N., and Gurevych, I. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, EMNLP.

[10] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed Representations of Words and Phrases and their Compositionality. Accepted to NIPS 2013.

[11] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pages 1532–1543, 2014.

[12] Yang Liu. Fine-tune BERT for extractive summarization. arXiv preprint arXiv:1903.10318, 2019.

[13] D. Cer, M. Diab, E. Agirre, I. Lopez-Gazpio, and L. Specia. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. arXiv preprint arXiv:1708.00055, 2017.

Machine Learning NLP Deep Learning Naturallanguageprocessing Transfer Learning