Sentence Representation Learning

Exercise 8

Fabian David Schmidt

Q1.1: Explain the training objective of the original Sentence-BERT transformer. Why does the objective enable cosine similarity search at inference time?
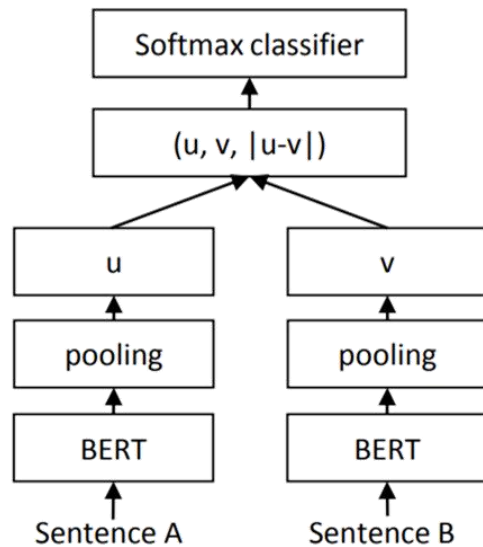

Figure 1: SBERT architecture with classification objective function, e.g., for fine-tuning on SNLI dataset. The two BERT networks have tied weights (siamese network structure).
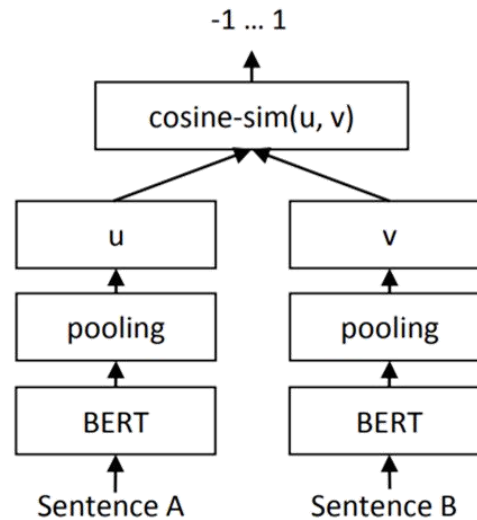

Figure 2: SBERT architecture at inference, for example, to compute similarity scores. This architecture is also used with the regression objective function.

- u, v are sentence representations of sentence pair
- Softmax objective trained on NLI linearly separates $(u, v, |u-v|)$ into entailment, contradiction, neutral
- Linear separation into classes closely related to angle of canonical class representation (i.e, each class vector in classifier)
- Classes align well with idea of sentence-level semantics
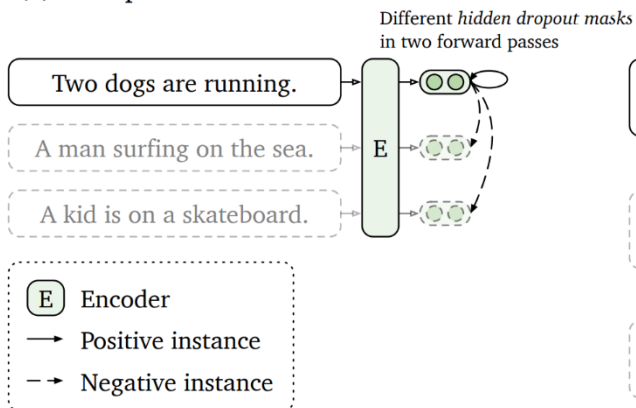- Good downstream (e.g., semantic search) representations

Q1.2: Can you think of intuitions as to why SRoBerta does not outperform SBERT, in contrast to other types of downstream tasks?

- BERT pre-trained with Masked Language Modelling and Next Sentence Prediction objectives
- RoBERTa only trained with Masked Language Modelling
- Neither of the two pretrains on sentence-level semantics very well, esp. on mean-pooled representations of token as a sentence embedding
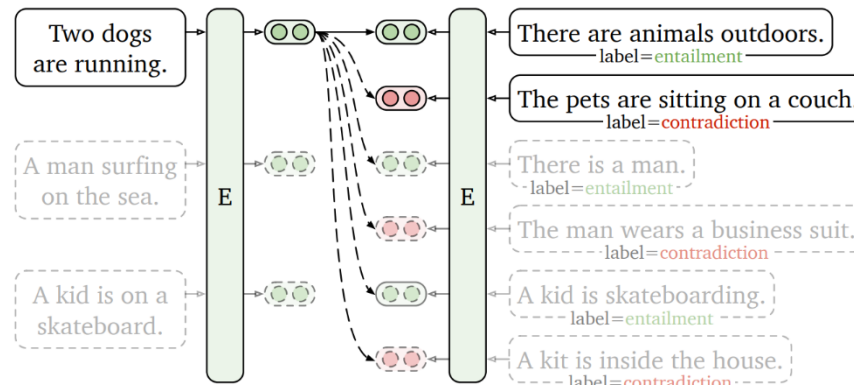
Q2.1: Briefly explain the core idea of contrastive learning and how the training objective is typically constructed.

(a) Unsupervised SimCSE

Different *hidden dropout masks* in two forward passes

Two dogs are running.

A man surfing on the sea.

A kid is on a skateboard.

E Encoder
→ Positive instance
⇢ Negative instance

(b) Supervised SimCSE

Two dogs are running.

A man surfing on the sea.

A kid is on a skateboard.

There are animals outdoors. label=entailment
The pets are sitting on a couch. label=contradiction
There is a man. label=entailment
The man wears a business suit. label=contradiction
A kid is skateboarding. label=entailment
A kit is inside the house. label=contradiction

Figure 1: (a) Unsupervised SimCSE predicts the input sentence itself from in-batch negatives, with different hidden dropout masks applied. (b) Supervised SimCSE leverages the NLI datasets and takes the entailment (premise-hypothesis) pairs as positives, and contradiction pairs as well as other in-batch instances as negatives.
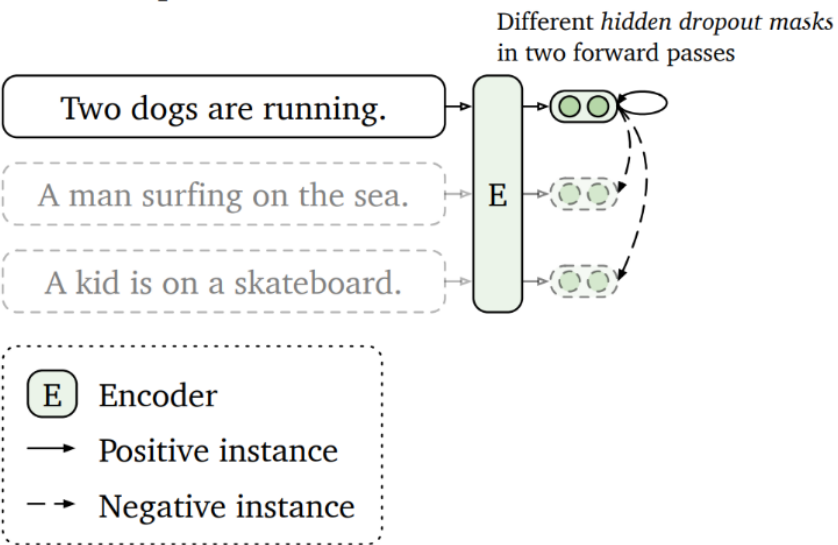
- **Core idea:** attract positive instances closer in representation space, repel negative instances
- **Loss:** softmax over cosine similarity typically in batch expressed as "multi-class classification" with 1 to k positive examples, all other in-batch instances are negatives
- **Considerations:** how to treat more than 1 positive, batch size (the larger the better!), multi-GPU training (where to put examples, other objectives, etc.)

Self-Supervised Representation Learning (2/3)

Q2.2: How does unsupervised SimCSE learn sentence-level representations in a selfsupervised fashion? How does it thereby improve over other potentially selfsupervised objectives?

(a) Unsupervised SimCSE

Different *hidden dropout masks* in two forward passes

Two dogs are running.

A man surfing on the sea.

E

A kid is on a skateboard.

E Encoder
→ Positive instance
⇢ Negative instance

Data augmentation			STS-B
None (unsup. SimCSE)			**82.5**
Crop	*10%*	*20%*	*30%*
	77.8	71.4	63.6
Word deletion	*10%*	*20%*	*30%*
	75.9	72.2	68.2
Delete one word			75.9
w/o dropout			74.2
Synonym replacement			77.4
MLM 15%			62.2

Table 1: Comparison of data augmentations on STS-B development set (Spearman's correlation). *Crop k%*: keep 100-k% of the length; *word deletion k%*: delete k% words; *Synonym replacement*: use `nlpaug` (Ma, 2019) to randomly replace one word with its synonym; *MLM k%*: use BERT$_{base}$ to replace k% of words.

- **Unsupervised SimCSE:** positive pair are repeated forward passes of the **same** instance, negatives are all other sentence within a batch
- **Repeated forward pass** results in very different sentence embeddings since initial output is highly misaligned and dropout masks meaningfully distort output
- **Other strategies** (cropping, word deletion, MLMing) are destructive in semantics to potentially align output incorrectly

Q2.3: . Imagine you want to train your own multilingual sentence transformer. List and briefly explain some key considerations in scaling up the training procedure.

- **Training objective:** typically some variant of contrastive loss, but maybe should also include language modelling (MLM, TLM) objectives
- **Training data:** large scale monolingual and parallel (bi- or n-way multilingual data)
- **Architecture:** sentence embedding models are typically not exorbitantly large; 12 to 24 layers should suffice
- **Tokenizer:** large-scale multilingual models should probably allocate a large capacity into the number of tokens (250-750K); trend goes towards larger vocabularies (varying scripts in multilinguality, programming languages, etc.)

Knowledge Distillation

Q3.1: What is knowledge distillation and how does it work (on the case of multilingual sentence transformers)?

Teacher Model Typically is a English sentence transformer trained on large-scale data

Multilingual model, e.g. XLM-Roberta
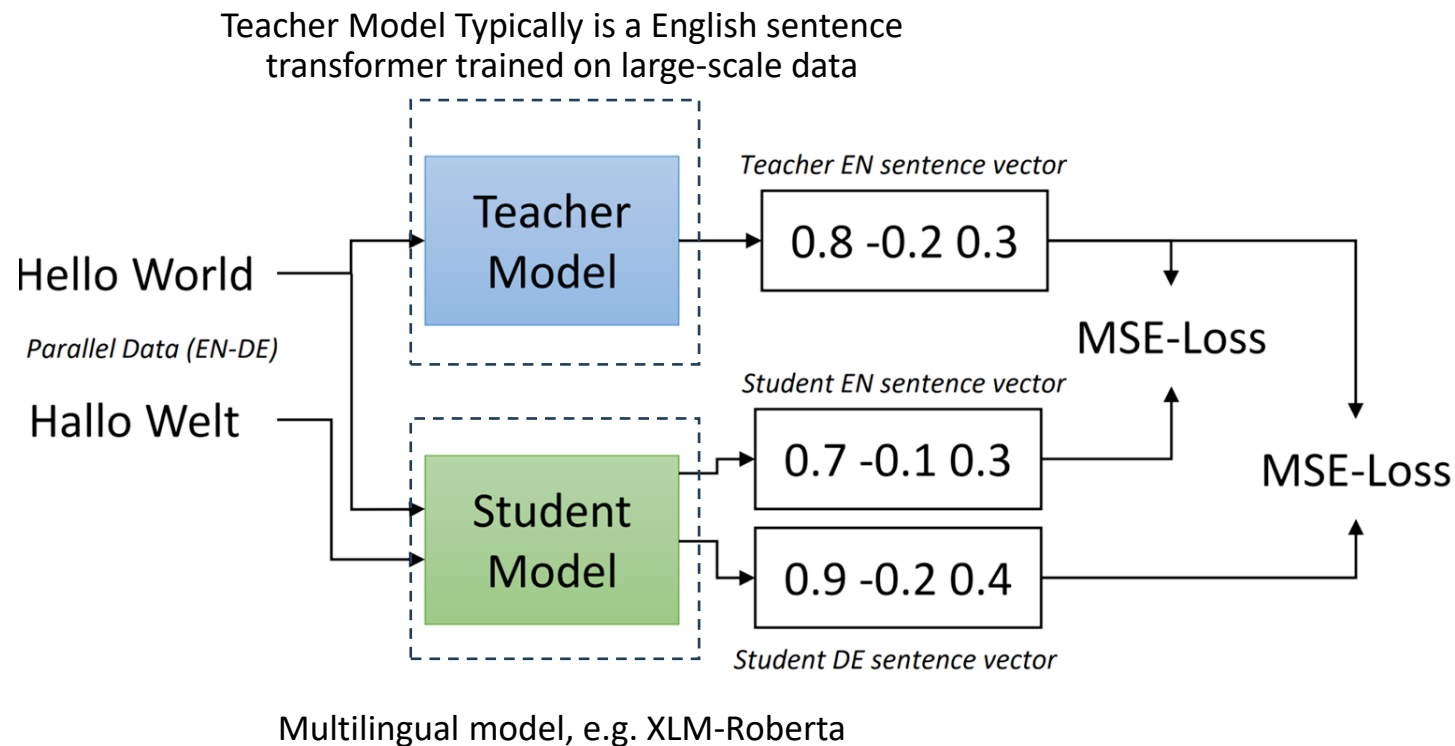
- **Core idea:** we re-lever sentence alignment of a pre-trained sentence embedder (teacher model) to align or multilingual model on parallel data
- **Parallel data:** sentence translations that are guaranteed to be semantically aligned
- **Objective:** MSE loss to minimize distance between teacher and student embeddings; other variants, e.g, on cosien similarity also conceivable
- **Q3.2:** quality of teacher and amount of data most critical – we can „only" replicate teacher and do so in best possible fashion