Multilingual NLP

10. Sentence Encoders
(+ Contrastive learning & Knowledge distillation)

Prof. Dr. Goran Glavaš
Center for AI and Data Science (CAIDAS), Uni Würzburg

Image: Alexander Mikhalchyk

After this lecture, you'll...

- Know how to obtain sentence-level encoders
- Understand how to train multilingual sentence encoders
- Be able to explain what contrastive learning is
- Understand what knowledge distillation is

Content

- **Sentence Embeddings**
- (Multilingual) Sentence Encoders
  + Contrastive Learning
  + Knowledge Distillation
- Self-Supervised Sentence Encoders

Sentence Embeddings

- **Sentence embedding** is a semantic representation of a sentence – a vector that captures the (fine-grained) meaning of the sentence
- Q: What do we need sentence embeddings for?
- Q: Can't we simply aggregate (e.g., average) sentence embeddings from word embeddings?

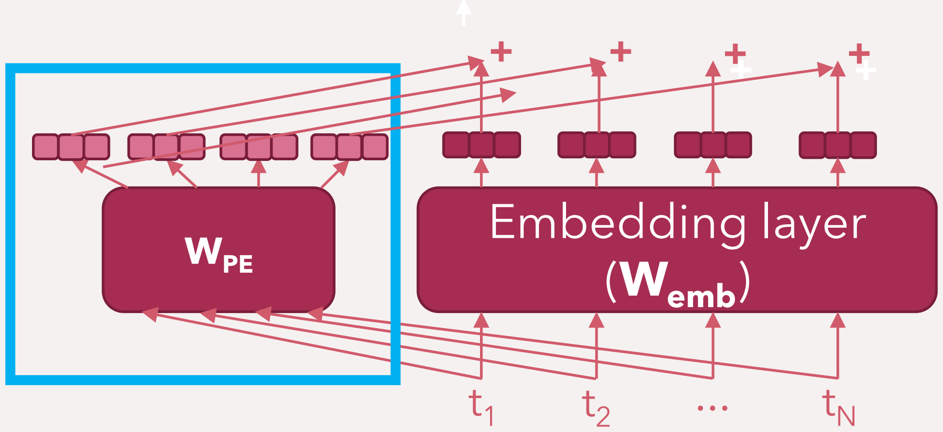
Sentence Embeddings

- **Sentence embedding** is a semantic representation of a sentence – a vector that captures the (fine-grained) meaning of the sentence
- Q: What do we need sentence embeddings for?
  - Information retrieval (find the sentence with the closest meaning)
  - Parallel data (bitext) mining (training data for MT)
  - Computationally efficient supervised training for sentence-level tasks
  - Data efficient supervised training for sentence-level tasks
- Q: Can't we simply aggregate (e.g., average) sentence embeddings from word-embeddings?
  - We can, but we won't get very good sentence representations
  - *the dog bit the man* vs. *the man bit the dog*?

Positional Embeddings

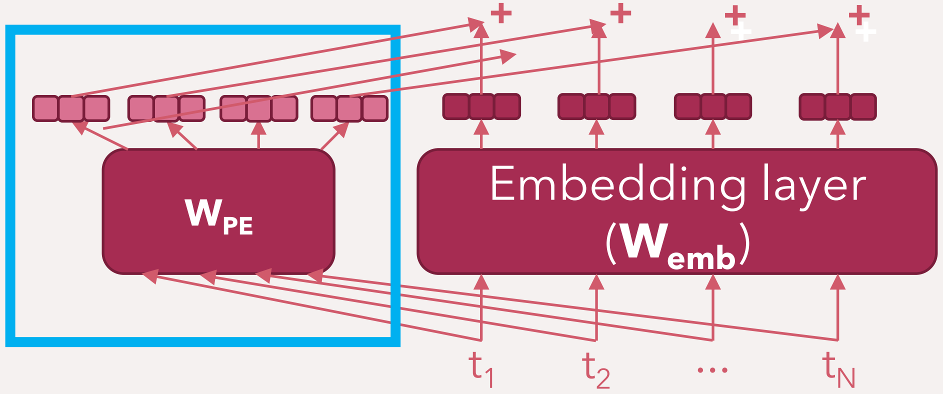
- But with pretrained Transformers, we have contextualized embeddings
- And these embeddings also encode the positions of tokens

- We also have special start tokens (e.g., [CLS] in BERT) that attend over all input tokens
- Q: just average token representations, output of the Transformer?
- Q: just take the output representation of the [CLS] token?



Positional Embeddings

- Q: just average token representations, output of the Transformer?
- Q: just take the output representation of the [CLS] token?
- Neither is <u>good enough</u> out of the box!
- Pretrained LMs are trained for predicting <u>(masked) words</u>, *not* encoding sentences!
  - Additional sentence-level training needed
  - Think of it as fine-tuning of a PLM to produce <u>sentence-level embeddings</u>



Training Sentence Encoders

- Obtaining a **sentence-level encoder** (i.e., a neural model that produces meaningful embeddings for input sentences) requires sentence-level training tasks/objectives
- Q: What could such task be?
  - It needs to require some kind of semantic comparison between two (or more) sentences
  - Ideally, requires fine-grained (precise) modeling of sentence meaning (small changes in wording can change the meaning much)
- Some tasks that fit:
  - Semantic text similarity (STS), Natural Language Inference (NLI), Paraphrase detection (PD)
  - Multilingually: translation detection – predict if sentences are translations of each other; Q: why not MT?
  - Q: where to get the data/annotations from?

Content

- Sentence Embeddings
- **(Multilingual) Sentence Encoders**
  - Contrastive Learning
  - Knowledge Distillation
- Self-Supervised Sentence Encoders

Training Sentence Encoders

Conneau, A., Kiela, D., Schwenk, H., Barrault, L., & Bordes, A. (2017, September). Supervised learning of universal sentence representations from natural language inference data. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (pp. 670-680). Association for Computational Linguistics.

- InferSENT: „pretrains" a (recurrent) sentence encoder on NLI
  - NLI is the sentence-pair task with largest training data
  - The same parametrized encoder Enc(s|θ) independently encoders both sentences
    - This architecture is often called Bi-Encoder or Siamese Network
  - Embeddings: **u** = Enc(s_1|θ), **v** = Enc(s_2|θ)
  - Concatenation of **u**, **v**, their absolute elementwise difference |**u**-**v**|, and their element-wise product **u***v**
    - Fed into the feed-forward softmax classifier
    - Predicts one of three NLI classes
    - Cross-entropy loss

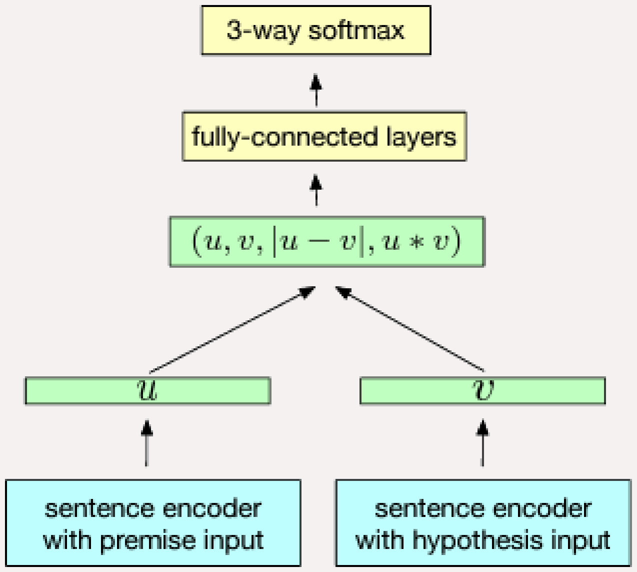
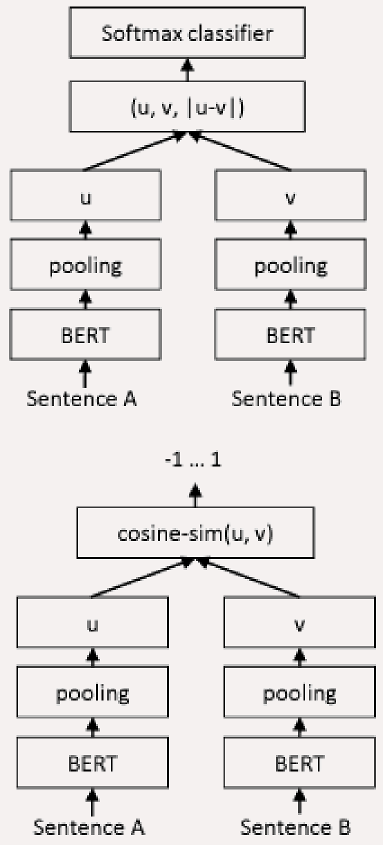


Image from the paper

Sentence-BERT

Reimers, N., & Gurevych, I. (2019, November). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 3982-3992).

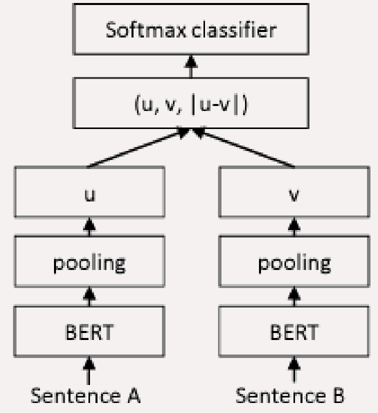
- SentenceBERT: fine-tuning BERT into a sentence-level encoder
- Trained on
  - NLI (classification objective)
    - Effectively the same as InferSent
    - But initialization with pretrained BERT (as opposed to random init of a deep Bi-LSTM in InferSent)
  - STS (regression objective)
    - Cosine between sentence embeddings compared against the human similarity score
      - Loss: mean square error

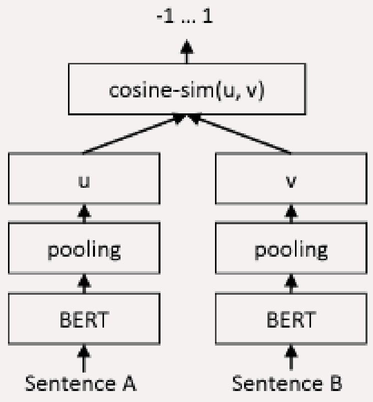


Sentence-BERT

Reimers, N., & Gurevych, I. (2019, November). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 3982-3992).

- SentenceBERT: supervised fine-tuning of BERT to obtain a sentence-level SBERT encoder
- But the large-scale data exists only in English
  - Primarily the NLI data
- Q: Can we obtain a multilingual sentence encoder the same way? How?
- Q: Fine-tune mBERT or XLM-R on the English NLI data?
  - Worse sentence embeddings for English compared to fine-tuning monolingual English PLMs
  - Bad performance for other languages due to large English-only fine-tuning





Multilingual Sentence Encoders

- Q: Which type of data do we generally have that aligns sentences across languages by meaning?
  - Parallel data (remember Lecture 9 and MT :)!
- Much of the work on training multilingual sentence encoders relies on (large amounts of) parallel data
- Two main approaches:
  1. Pretraining from scratch (or from general-purpose PLMs)
     - Typically with **contrastive learning** objectives
  2. Transfering sentence-encoding knowledge from a monolingual sentence encoder to a multilingual PLM
     - Typically via **knowledge distillation**

Multilingual Sentence Encoders

Artetxe, M., & Schwenk, H. (2019). Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. Transactions of the Association for Computational Linguistics, 7, 597-610.

LASER: „Universal language agnostic sentence embedder"

- Pretraining from scratch:
- Massively multilingual: 93 languages
- Recurrent sentence encoder (Bi-LSTM) trained via encoder-decoder NMT

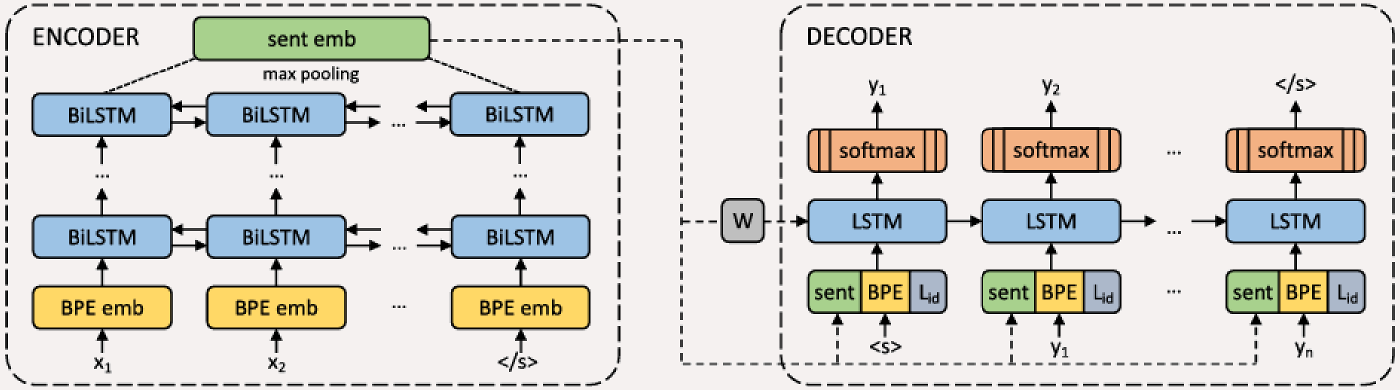


Image from the paper

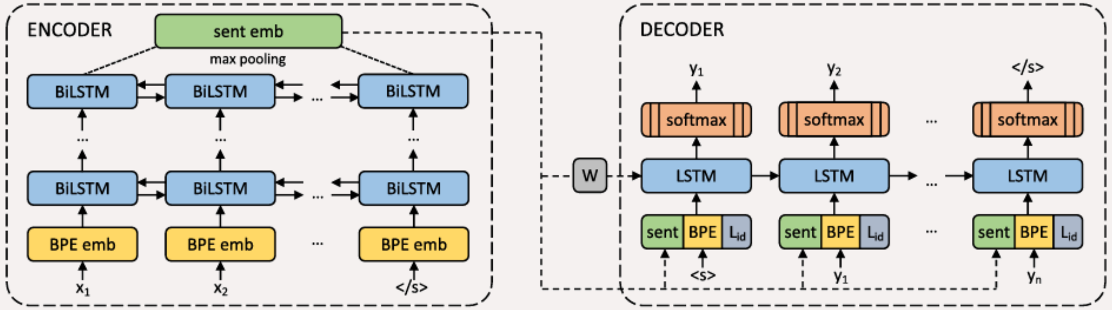
Multilingual Sentence Encoders

Artetxe, M., & Schwenk, H. (2019). [Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. Transactions of the Association for Computational Linguistics](), 7, 597-610.

LASER: „Universal language agnostic sentence embedder"

- Sentence embedding (semb): max-pooled token-level representations, output of the last Bi-LSTM layer of the encoder
- Initial state of the decoder = sentence embedding linearly projected (matrix **W**)
- Sentence embeddings also fed as input to decoder at each step
- Trainable language ID vectors (also at input to decoder)
- Shared BPE vocabulary across the 93 languages

Image from the paper



Content

- Sentence Embeddings
- **(Multilingual) Sentence Encoders**
  - + **Contrastive Learning**
  - + Knowledge Distillation
- Self-Supervised Sentence Encoders

Multilingual Sentence Encoders

Chidambaram, M., Yang, Y., Cer, D., Yuan, S., Sung, Y., Strope, B., & Kurzweil, R. (2019, August). Learning Cross-Lingual Sentence Representations via a Multi-task Dual-Encoder Model. In Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019) (pp. 250-259).

mUSE: „Multilingual Universal Sentence Embeddings"

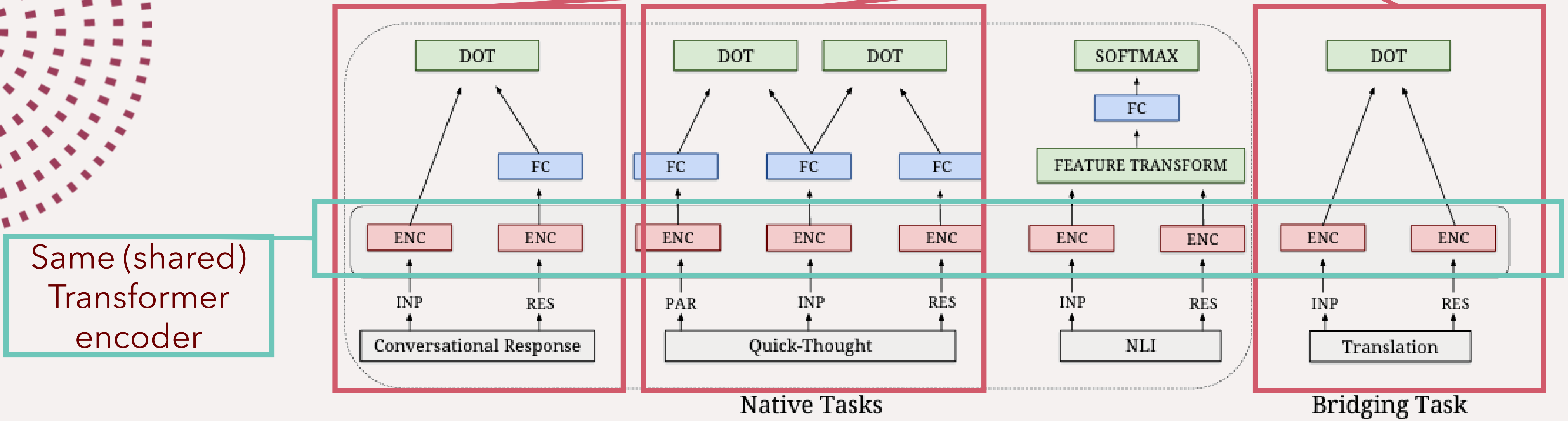
- Transformer encoder (trained from scratch, not a PLM)
- Train bilingual rather than multilingual encoders
- Multi-task training: 4 training tasks tasks
  - NLI – same as in InferSent and SBERT
  - Other are ranking tasks with a contrastive objective
  - 3 tasks are actually monolingual English
- Cross-lingual sentence ranking
  - For an input sentence in English, score the target language sentences (translation should be on top of the ranking)

Multilingual Sentence Encoders

Chidambaram, M., Yang, Y., Cer, D., Yuan, S., Sung, Y., Strope, B., & Kurzweil, R. (2019, August). Learning Cross-Lingual Sentence Representations via a Multi-task Dual-Encoder Model. In Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019) (pp. 250-259).

mUSE: „Multilingual Universal Sentence Embeddings"

- Transformer encoder (trained from scratch, not a PLM)
- Other are ranking tasks with a contrastive objective



Contrastive Learning

- **Contrastive learning** is a learning paradigm that trains some model (parameters) by constrasting (comparing) two or more instances
- In traditional learning objectives, the model sees each instance independently (and makes some prediction for the instance)

$$\boldsymbol{loss_{tr}}(model(\mathbf{x}|\boldsymbol{\theta}), y)$$

- In contrastive learning, the loss is a function based on comparison of model's predictions for two or more instances

$$\boldsymbol{loss_{ctr}}(\boldsymbol{cmp}(model(\mathbf{x}_1|\boldsymbol{\theta}), model(\mathbf{x}_2|\boldsymbol{\theta}), \ldots, model(\mathbf{x}_n|\boldsymbol{\theta})))$$

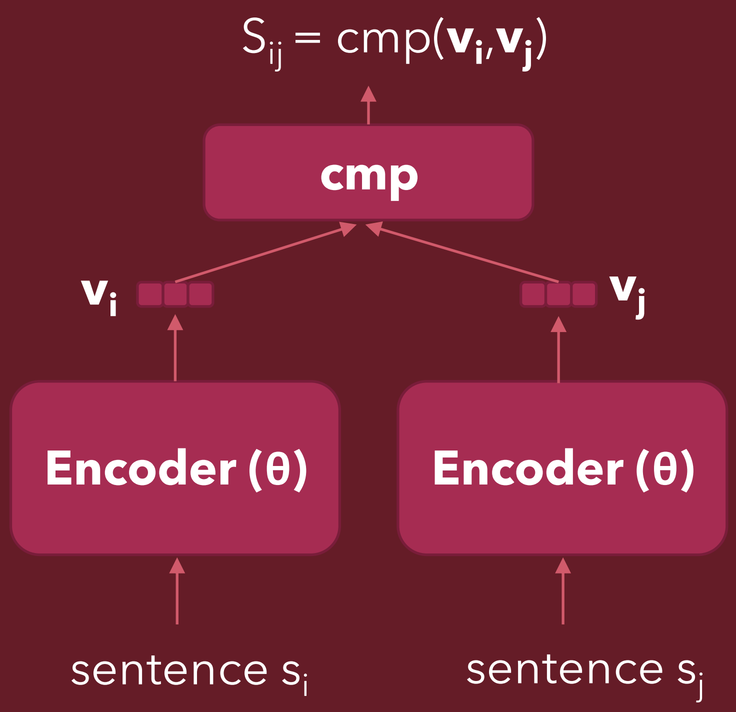
- Cmp is the function that compares the model outputs for instances
- Closely related to the concept of metric learning

Contrastive Learning

- **Bi-Encoder** (or Siamese Network) is common architecture on top of which contrastive learning objectives are applied
- Two instances are fed independently to the encoder

$$\mathbf{v_i} = \text{Encoder}(s_i|\boldsymbol{\theta})$$
$$\mathbf{v_j} = \text{Encoder}(s_j|\boldsymbol{\theta})$$

- **cmp** is then some distance of similarity metric that compares $\mathbf{v_i}$ and $\mathbf{v_j}$
  - Typically non-parameterized!
  - Euclidean dist. or cosine sim.

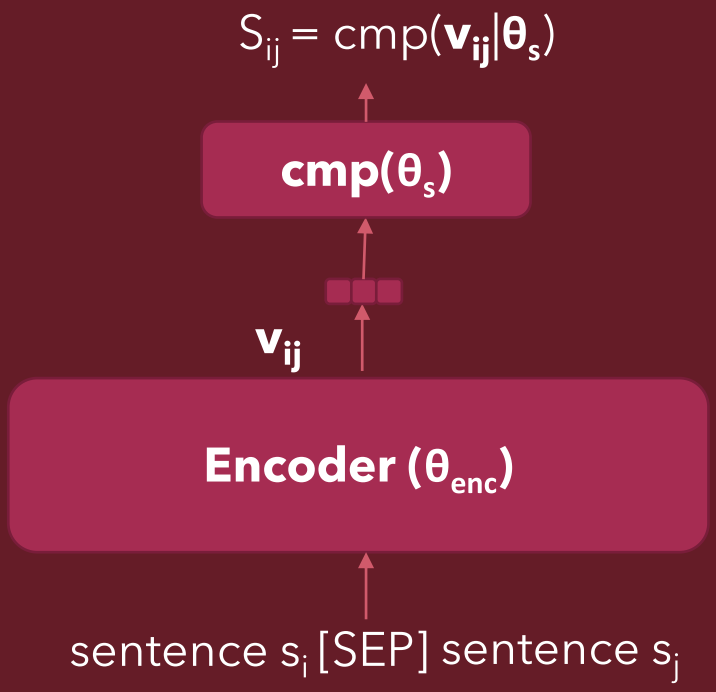


Contrastive Learning

- In the Bi-Encoder architecture, each sentence is encoded <u>independently</u> – encoding of each sentence does not reflect the other
  - Can be quite important in some tasks (e.g., NLI)
- Alternative: **Cross-Encoder** architecture
  - Pair of sentences encoded together
  - Concatenated at input to the encoder

$$\mathbf{v}_{ij} = \text{Encoder}(s_i + s_j | \theta_{enc})$$

- Encoder outputs a joint representation for a sentence pair (s_i, s_j)
- Must have a parameterized (θ_s) scoring function cmp to convert $\mathbf{v}_{ij}$ into a scalar score
  - <u>Cannot obtain embeddings of individual sentences</u>



Contrastive Learning

- Contrastive loss functions contrast the scores s_{ij} of positive pairs (i, j) against the scores s_{ik} (or s_{jk}) of negative pairs (i, k)
- **Positive pair**: pairs of instances (sentences) for which the relation of interest holds, e.g., s_i and s_j are mutual translations
- **Negative pairs**: pairs made of one instance from the positive pair (e.g., s_i) and instances s_k such that the relation that holds for (s_i, s_j) does not hold for (s_i, s_k) (e.g., s_k is not a translation of s_i)
  - We typically contrast (sometimes simultaneously) the same positive pair against multiple negatives

Contrastive Learning

- **Triplet loss** – arguably the simplest contrastive loss:
  - Positive (s_i, s_j) with the score s_{ij} contrasted against a single negative (s_i, s_k) with the score s_{ik}

$$loss_{triplet}(s_i, s_j, s_k) = \max(0, s_{ik} - s_{ij} + \varepsilon)$$

    - This formulation assumes that s is a similarity score
    - If s is supposed to be a distance score, then $\max(0, s_{ij} - s_{ik} + \varepsilon)$

  - Pushes the model to score positive pairs higher than corresponding negative pairs by (at least) a margin ε
  - By minimizing this loss, the encoder must „figure out" what sentences of positive pairs share that those of negative pairs don't

Contrastive Learning

Oord, A. V. D., Li, Y., & Vinyals, O. (2018). Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748.

- Noise Contrastive Estimation (**InfoNCE**): contrast simultaneously the positive pair (s_i, s_j) against N negatives (s_i, s_{k1}), (s_i, s_{k1}), ..., (s_i, s_{kN})

$$\text{loss}(s_i, s_j, s_{k1}, \ldots, s_{kN}) = -\ln \frac{\exp\left(\frac{s_{ij}}{T}\right)}{\exp\left(\frac{s_{ij}}{T}\right) + \sum_{k_1}^{k_N} \exp\left(\frac{s_{ik}}{T}\right)}$$

- Effectively negative log-likelihood of a positive pair, according to a softmax over all scores: for the positive and all negatives
- **T** = temperature hyperparam. of the loss – controls the „strength" of the contrast, i.e., how much we smoothen the raw scores s_{ij}
  - Q: What if $T = 0$? What if $T = \infty$?

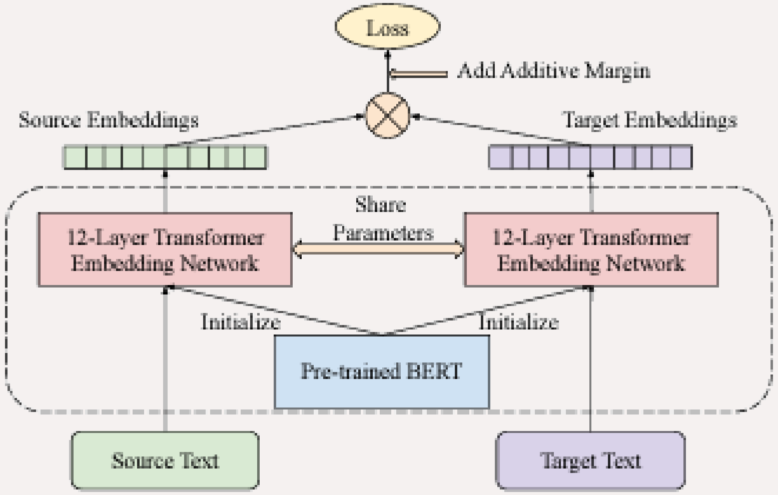
Contrastive Learning

- Positives are typically given in the data we train on
  - E.g., in parallel data – sentences that are translations of each other
- Q: But how do we create negatives?
  - Fixed, predetermined negatives for each positive? Or
  - Dynamically selected within the training batch of the positive?
- Both strategies are used
  - In-batch negatives (aka <u>random</u> (but not fixed) negatives)
  - Fixed, precomputed negatives are selected based on some criteria when we need hard negatives
    - Hard negatives: in some aspect somewhat similar to positives
    - E.g., s_k not a translaton of s_i, but shares some meaning (i.e., not completely semantically unrelated)

Multilingual Sentence Encoders

Feng, F., Yang, Y., Cer, D., Arivazhagan, N., & Wang, W. (2022, May). Language-agnostic BERT Sentence Embedding. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 878-891).

- LABSE: „Language-Agnostic BERT Sentence Embedding"
  - Basically mUSE, but the Encoder is initialized with mBERT
  - Cross-lingual sentence ranking using parallel data (100M pairs for each lang!)
    - Modified InfoNCE loss – what they call „additive margin"
    - $\exp(s_{ij} - m)$ instead of $\exp(s_{ij}/T)$ for the pos and only $\exp(s_{ij})$ for the negs
      - „Improves separation between translations and near non-translations"



Content

- Sentence Embeddings
- **(Multilingual) Sentence Encoders**
  + Contrastive Learning
  + **Knowledge Distillation**
- Self-Supervised Sentence Encoders

Multilingual Sentence Encoders

Reimers, N., & Gurevych, I. (2020, November). Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 4512-4525).

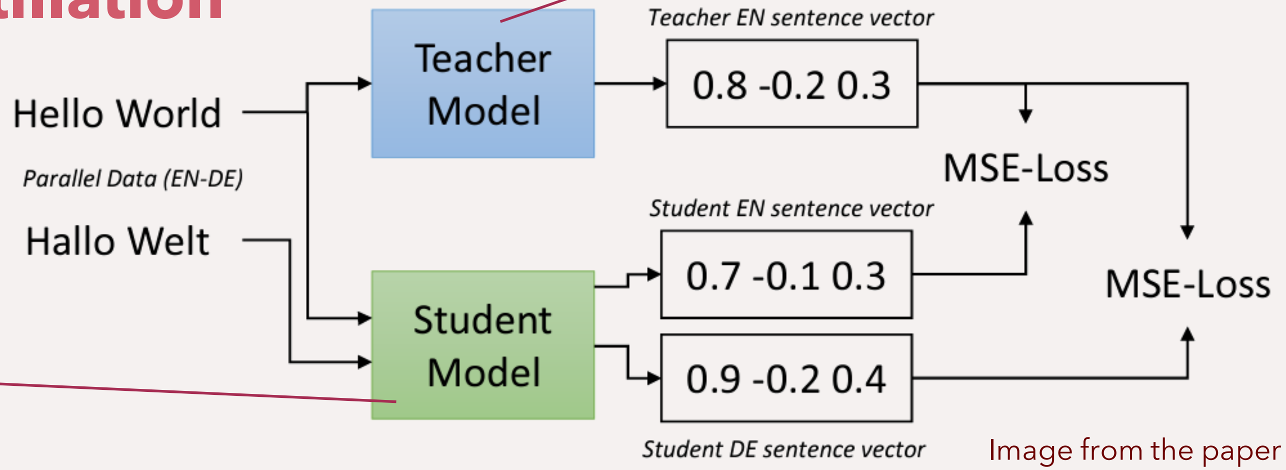
- Parallel data allows for pretraining multilingual sentence encoders
  - Especially if we start from multilingual MMTs like mBERT
- Monolingual English SBERT:
  - Supervised fine-tuning: NLI, STS, …
  - Stronger (for EN) than multilingual SEs trained from scratch with parallel data
- Idea: „copy" the sentence specialization knowledge of English SBERT into a multilingual MMT like mBERT
  - Supervision for „copying": parallel data!

Multilingual Sentence Encoders

Reimers, N., & Gurevych, I. (2020, November). Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 4512-4525).

- Idea: „copy" the sentence specialization knowledge of English SBERT into a multilingual MMT like mBERT
  - Supervision for „copying": parallel data!
- „Copying" = **knowledge distillation**

English Sentence-BERT



Initialized with an MMT (mBERT or XLM-R)English SBERT

Image from the paper

Knowledge Distillation

Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531.

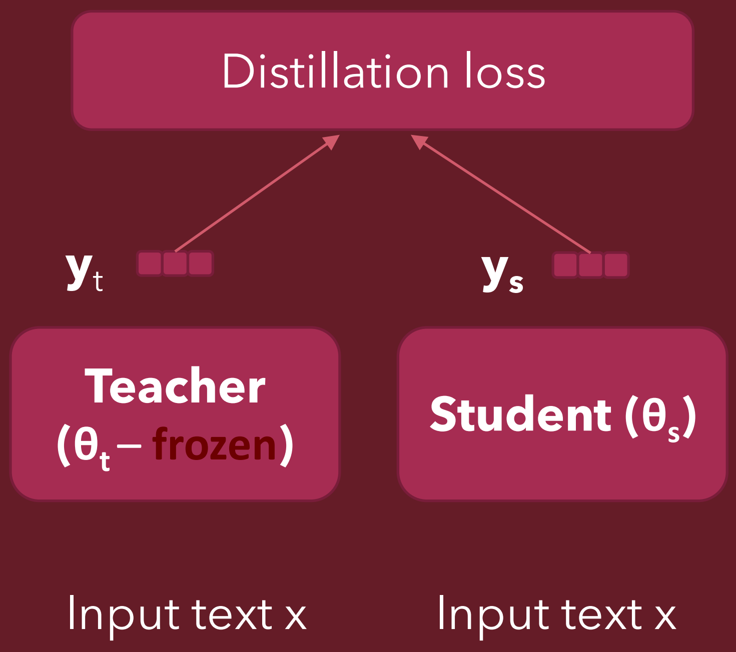
- **Knowledge distillation** is a process of replicating the „knowledge" stored in the parameters of one model – the **teacher** – into the parameters of a new model – the **student**

- Q: Why would we do knowledge distillation?
  - Original work used it to distill multiple models for the same task (an ensemble of models) into a single model
  - Compression – distilling a larger model into a smaller model
    - Pretrained LMs have been shown to be overparametrized!

Knowledge Distillation

Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531.

- **Knowledge distillation**: we need some training data for distillation too
  - We feed the same input x to both the teacher and student
- $\mathbf{y}_t$ and $\mathbf{y}_s$ outputs of teacher and student, resp.
  - E.g., an encoding of the sentence
  - E.g., a probability distribution over classes
- Training loss: minimize the difference between the student output and teacher output
  - E.g., minimize Euclidean distance or maximize cosine similarity between $\mathbf{y}_t$ and $\mathbf{y}_s$



Multilingual Sentence Encoders

Reimers, N., & Gurevych, I. (2020, November). Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 4512-4525).

- Sentence-BERT: specialized for encoding sentences in English
  Supervision for „copying“: parallel data!
- mBERT: „understands“ input text in 104 languages
- Distill the knowledge of EN SBERT into mBERT
  - We get a multilingual Sentence-BERT!

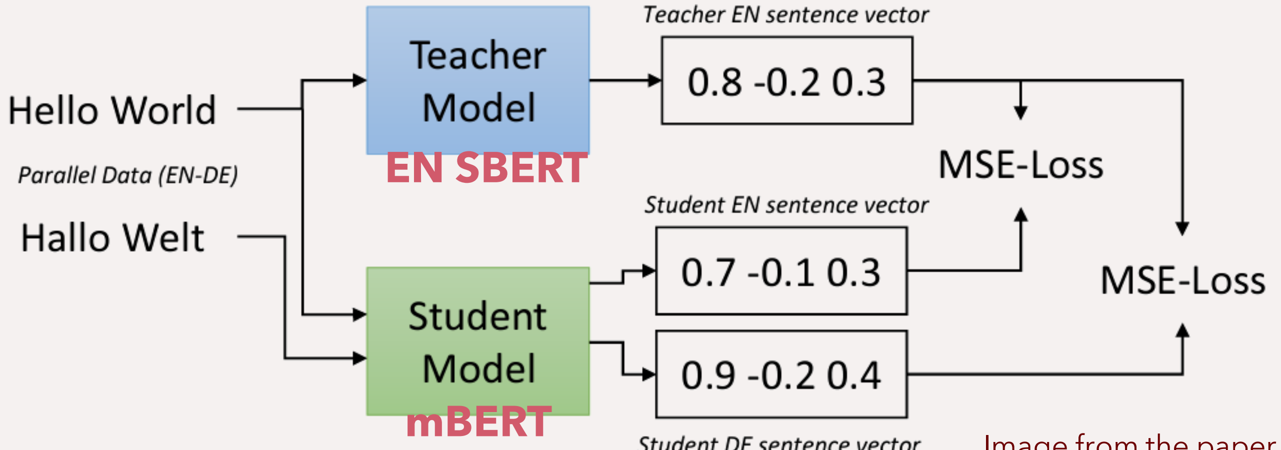


Image from the paper

Multilingual Sentence Encoders

Reimers, N., & Gurevych, I. (2020, November). Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 4512-4525).

- Distill the knowledge of EN SBERT into mBERT
- Parallel data needed:
  - (s_{EN}, s_{L2}) – one translation pair
  - s_{EN} encoded by both the teacher ($\mathbf{s}^t_{EN}$) and student ($\mathbf{s}^s_{EN}$)
  - s_{L2} encoded by the student ($\mathbf{s}_{L2}$)
  - Distillation loss:

$$MSE(\mathbf{s}^t_{EN}, \mathbf{s}^s_{EN}) + MSE(\mathbf{s}^t_{L2}, \mathbf{s}^s_{EN})$$

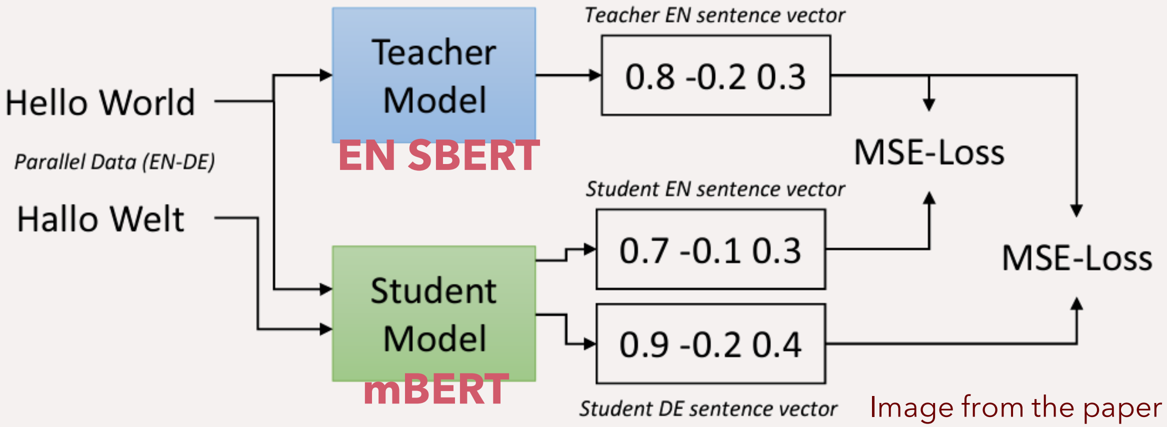


Image from the paper

Multilingual Sentence Encoders

Heffernan, K., Çelebi, O., & Schwenk, H. (2022, December). Bitext Mining Using Distilled Sentence Representations for Low-Resource Languages. In Findings of the Association for Computational Linguistics: EMNLP 2022 (pp. 2101-2112).

- All previous multilingual approaches train the same encoder (all parameters shared) for all languages
    - Positive transfer across languages vs. the curse of multilinguality?
- „LASER3": distills a massively multilingual LASER teacher into many student models, one for each language or lang. group
    - Each student with its own language-specific SentPiece tokenizer
    - Since all students are distilled from the same teacher, their output embeddings are comparable (in the same representation space)

Multilingual Sentence Encoders

Heffernan, K., Çelebi, O., & Schwenk, H. (2022, December). Bitext Mining Using Distilled Sentence Representations for Low-Resource Languages. In Findings of the Association for Computational Linguistics: EMNLP 2022 (pp. 2101-2112).

- „LASER3": distills a massively multilingual LASER teacher into many student models, one for each language or language group
  - Teacher: a recurrent model (LASER)
  - Students: 12-layer Transformers
  - Distillation: similar to Reimers & Gurevych
    - Only maximizing cosine similarity instead of minimizing MSE
  - Student additionally trained monolingually on L2 data, via MLM-ing

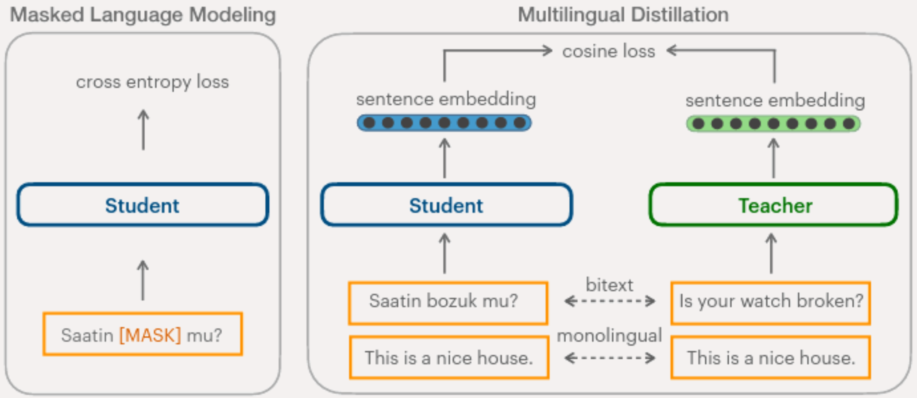


Image from the paper

Content

- Sentence Embeddings
- (Multilingual) Sentence Encoders
  + Contrastive Learning
  + Knowledge Distillation
- **Self-Supervised Sentence Encoders**

Multilingual Sentence Encoders

- All sentence encoders we introduced so far needed some kind of annotated data for supervision
  - Monolingual EN SBERT: NLI and STS
  - Multilingual SBERT: both NLI/STS (for the teacher) + parallel data (for the distillation)
  - LASER, LaBSE, LASER3: parallel data
- Q: could we train sentence encoders in a self-supervised manner?
  - Like regular PLMs (BERT & co.) are trained via MLM-ing
  - Yes, via data augmentation!
    - For any sentence, we need to automatically create positive pairs
    - Create „paraphrases" (same meaning, different words)
      - In the symbolic or representation space

Multilingual Sentence Encoders

Gao, T., Yao, X., & Chen, D. (2021). SimCSE: Simple Contrastive Learning of Sentence Embeddings. In 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021 (pp. 6894-6910). Association for Computational Linguistics (ACL).

- SimCSE: for an input sentence s_i, obtain two different embeddings $\mathbf{s}_i^1$ and $\mathbf{s}_i^2$
  - Q: But how to obtain two different embeddings with the very same encoder?
    - We could add random noise to some intermediate representations in the Transformer, or
    - Dropout!

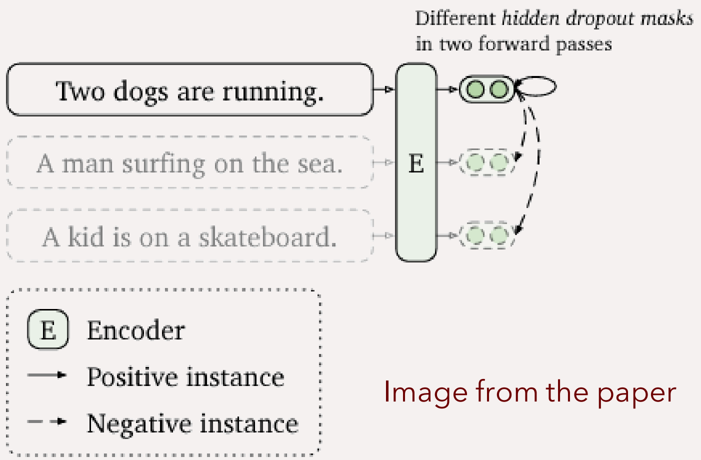


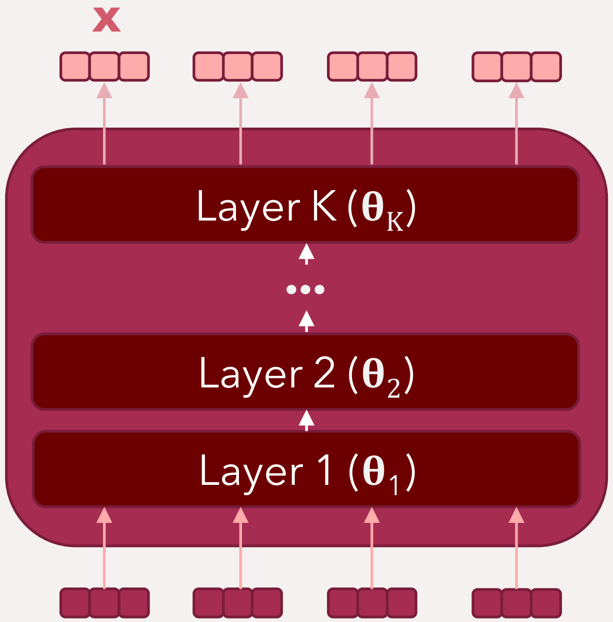
Image from the paper

Dropout

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The journal of Machine Learning Research, 15(1), 1929-1958..

- Let **x** be any hidden representation, output of any layer (e.g., in our neural LM)
  - E.g., output of layer K
- Applying dropout on a layer means
  - To modify its output(s) **x** so that each element x_i becomes replaced with x'_i:

$x'_i = 0$ with dropout probability p or
$x'_i = x_i / (1-p)$ with the probability $(1-p)$



Multilingual Sentence Encoders

Gao, T., Yao, X., & Chen, D. (2021). SimCSE: Simple Contrastive Learning of Sentence Embeddings. In 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021 (pp. 6894-6910). Association for Computational Linguistics (ACL).

- SimCSE: for an input sentence s_i, obtain two different embeddings $\mathbf{s}_i^1$ and $\mathbf{s}_i^2$
  - Same sentence passed through encoder twice, with two different dropout masks
  - The obtained embeddings $(\mathbf{s}_i^1, \mathbf{s}_i^2)$ then make a positive pair for contrastive training
  - Negative pairs: $(\mathbf{s}_i^1, \mathbf{s}_j)$ or $(\mathbf{s}_i^2, \mathbf{s}_j)$ where $\mathbf{s}_j$ is the embedding of some other sentence s_j

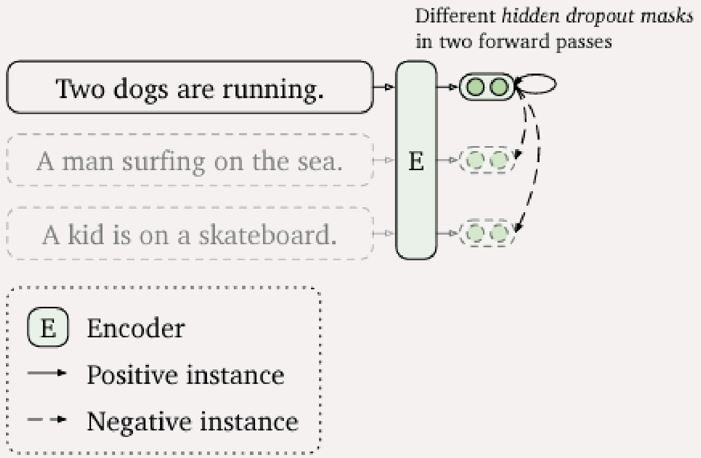


Image from the paper

Image: Alexander Mikhalchyk