Enhancing Multilingual Information Retrieval with Reranking using Mono-Language Sentence Embeddings and Knowledge Distillation

Saurabh Modi (B20EE035), Mukul Shingwani (B20AI023), Jaimin Gajjar (B20AI014)

Abstract

In this project, a multilingual reranking pipeline is being developed using Mono language sentence embeddings and knowledge distillation. The first step involves creating models for cross language sentence embeddings using knowledge distillation techniques. Then this sentence embedding models are used to create the Multilingual Biencoder structures which in turn as used for reranking.

1. Proposed Methodology

This pipeline aims to compare sentences in multiple languages (English, German, Arabic) with a query sentence in English using a biencoder approach. The biencoder approach involves using different types of biencoders (En-En, En-De, En-Ar) to compare sentences with their respective language pair. To ensure that the scores generated by the embedding generating BERT models are comparable, they are trained using knowledge distillation along with mean pooling. A multi language biencoder structure is as depicted in Figure 2.

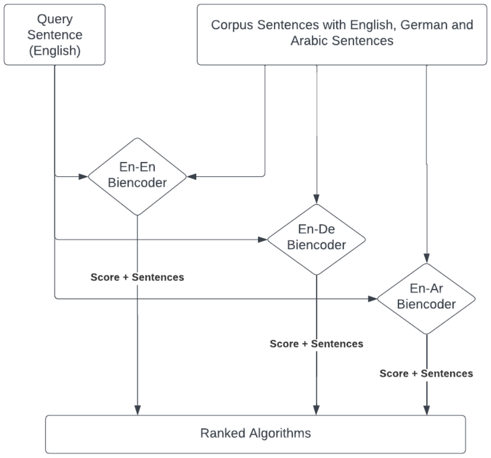


Figure 1. Multilingual Information Retrieval with Reranking

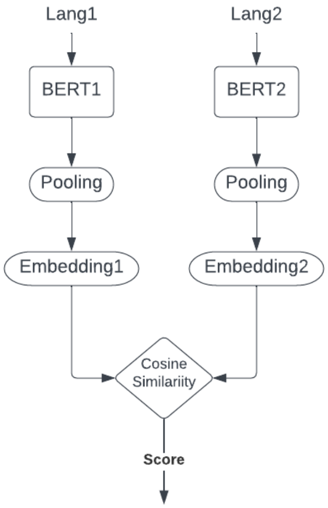


Figure 2. Multilingual Biencoder

2. Target Language Monolingual Sentence Embedding Models using Knowledge Distillation

The idea is based on a fixed (monolingual) teacher model, that produces sentence embeddings with our desired properties in one language. The student model is supposed to mimic the teacher model, i.e., the same English sentence should be mapped to the same vector by the teacher and by the student model. In order that the student model works for further languages, we train the student model on parallel (translated) sentences. The translation of each sentence should also be mapped to the same vector as the original sentence.

In the above figure, the student model should map Hello World and the German translation Hallo Welt to the vector of teacher model('Hello World'). We achieve this by training the student model using mean squared error (MSE) loss.

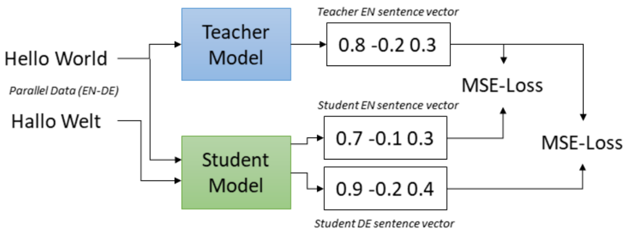
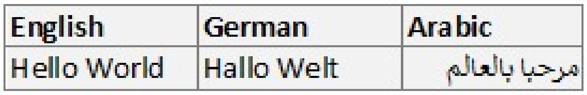


Figure 3. Extending to Target Language Monolingual Models

2.1. Dataset

To train the cross-language sentence embedding models, the TED2020 corpus is utilized, which contains transcripts and translations from TED and TEDx talks. The approach involves extending a monolingual model to several languages, including English, German, Spanish, Italian, French, Arabic, and Turkish. However, for this project, only the English, German, and Arabic parallel translated training and validation datasets are used.



English	German	Arabic
Hello World	Hallo Welt	مرحبا بالعالم

Figure 4. Data Format

2.2. Model Used

2.2.1 Teacher Model

We use an **English SBERT** model as teacher model M. Model Trained fully from scratch with evaluation done on TED2020 dataset.

2.2.2 Student Model

We use **XLM RoBERTa (XLM-R)** as student model $\hat{M}$. The English BERT models have a wordpiece vocabulary size of 30k mainly consisting of English tokens. Using the English SBERT model as initialization for $\hat{M}$ would be suboptimal, as most words in other latin-based languages would be broken down to short character sequences, and words in nonlatin alphabets would be mapped to the UNK token. In contrast, XLM-R uses SentencePiece2, which avoids language specific pre-processing. Further, it uses a vocabulary with 250k entries from 100 different languages. This makes XLM-R much more suitable for the initialization of the multilingual student model.

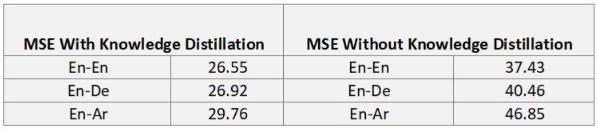
2.3. Training Settings

1. **Teacher Model:** English SBERT model
2. **Student Model:** XLM RoBERTa (XLM-R)
   (a) **Vocab size**: 250002
   (b) **Position embedding type:** Absolute
   (c) **Num hidden layers:** 12
   (d) **Num attention heads:** 12
   (e) **Max position embeddings:** 514
   (f) **Layer norm eps:** 1e-05
   (g) **Intermediate size:** 3072
   (h) **Initializer range:** 0.02
   (i) **Hidden size:** 768
   (j) **Hidden dropout prob:** 0.1
   (k) **Hidden activation:** GeLU
   (l) **Word embedding dimension:** 768

3. Evaluation Tests

3.1. MSE Evaluation

You can measure the mean squared error (MSE) between the student embeddings and teacher embeddings. This evaluator computes the teacher embeddings for the source sentences in English. During training, the student model is used to compute embeddings for the target sentences in German and Arabic respectively. The distance between teacher and student embeddings is measured. Lower scores indicate a better performance.



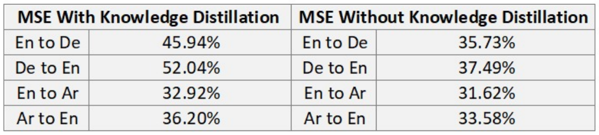
MSE With Knowledge Distillation		MSE Without Knowledge Distillation	
En-En	26.55	En-En	37.43
En-De	26.92	En-De	40.46
En-Ar	29.76	En-Ar	46.85

Figure 5. MSE Evaluation Results

3.2. Translation Accuracy

You can also measure the translation accuracy. Given a list with source sentences, for example, 1000 English sentences. And a list with matching target (translated) sentences, for example, 1000 German sentences.

For each sentence pair, we check if their embeddings are the closest using cosine similarity. I.e., for each source sentence we check if target sentence has the highest similarity out of all target sentences. If this is the case, we have a hit, otherwise an error. This evaluator reports accuracy (higher = better).



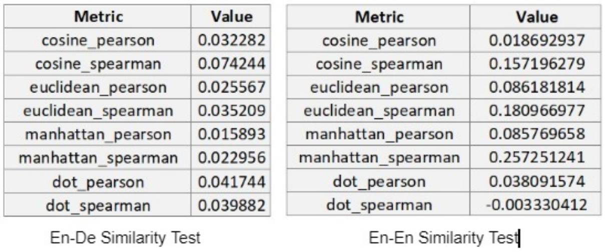
MSE With Knowledge Distillation		MSE Without Knowledge Distillation	
En to De	45.94%	En to De	35.73%
De to En	52.04%	De to En	37.49%
En to Ar	32.92%	En to Ar	31.62%
Ar to En	36.20%	Ar to En	33.58%

Figure 6. Translation Accuracy

3.3. Multi-Lingual Semantic Textual Similarity

You can also measure the semantic textual similarity (STS) between sentence pairs in different languages.

Where sentences1 and sentences2 are lists of sentences and the score is a numeric value indicating the semantic similarity between sentences1 item and sentences2 item.



Metric	Value
cosine_pearson	0.032282
cosine_spearman	0.074244
euclidean_pearson	0.025567
euclidean_spearman	0.035209
manhattan_pearson	0.015893
manhattan_spearman	0.022956
dot_pearson	0.041744
dot_spearman	0.039882

En-De Similarity Test

Metric	Value
cosine_pearson	0.018692937
cosine_spearman	0.157196279
euclidean_pearson	0.086181814
euclidean_spearman	0.180966977
manhattan_pearson	0.085769658
manhattan_spearman	0.257251241
dot_pearson	0.038091574
dot_spearman	-0.003330412

En-En Similarity Test

Figure 7. Semantic Similarity Test Results

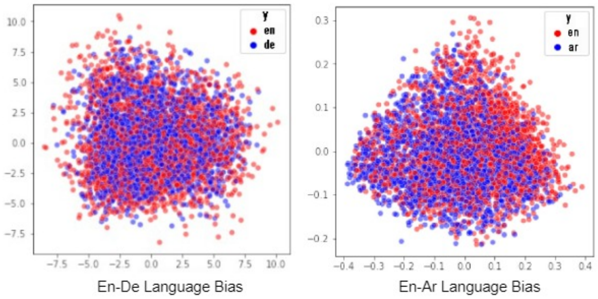
3.4. Language Bias

Language bias is a phenomenon where a model shows a preference for a particular language or language pair over others. For instance, a model may map sentences in the same language closer in vector space simply because they are of the same language. This can be problematic when dealing with multilingual sentence pools, as certain language pairs may be discriminated against, potentially harming the overall performance for multilingual data.

The figures below shows the plot of the first two principal components for different multilingual sentence embedding methods. In the plot, we encoded the English premise sentences from TED2020 with their German translation. The plot shows for the SBERT model a drastic separation between the two languages, indicating that the language significantly impacts the resulting embedding vector.

4. Limitations

- The use of Biencoders is only applicable to short and less number of documents. Cross-encoders are to be used if it is to be done for large number of documents. Cross-Encoder cannot be implemented via this pipeline as it does not work on mean pooling of the sentence embeddings to bring them on same vector space for preserve their semantic similarity.



En-De Language Bias

En-Ar Language Bias

Figure 8. Language Bias

- The drop of efficacy in student model as compared to teacher model may lead to creation of a bias on reranking algorithm on the parent language.

References

Supervised Learning of Universal Sentence Representations from Natural Language Inference Data
Supervised Learning of Universal Sentence Representations from Natural Language Inference Data
Unsupervised Cross-lingual Representation Learning at Scale
Language-agnostic BERT Sentence Embedding
Margin-based Parallel Corpus Mining with Multilingual Sentence Embeddings
Margin-based Parallel Corpus Mining with Multilingual Sentence Embeddings
Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation

5. Appendix

5.1. Using Cross Encoders along with Biencoders

Cross-Encoders are useful for computing similarity scores for pre-defined sentence pairs, while Bi-Encoders are better suited for generating sentence embeddings for applications such as information retrieval or clustering. Although Cross-Encoders have higher performance, they do not scale well for large datasets. To address this, a combination of both Cross- and Bi-Encoders can be used in scenarios such as information retrieval, where a Bi-Encoder is first used to retrieve similar sentences, and a Cross-Encoder is then used to re-rank the results.

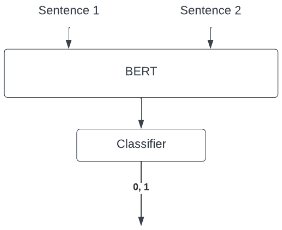


Figure 9. Cross Encoder Architecture