Notes on Sentence Embedding Models by Nils Reimers

compiled by D. Gueorguiev, 6/15/2025

# Preliminaries

## The Pooling Layer

The purpose of the pooling layer in both BERT and SBERT (Sentence-BERT) is to convert the variable-length sequence of token embeddings produced by the transformer into a fixed-size sentence embedding. This fixed-size representation is crucial for tasks like sentence similarity comparison, where you need to compare the "meaning" of different sentences represented as vectors.

The pooling layer in BERT and SBERT plays a crucial role in converting the model's output, which is a sequence of contextualized token embeddings, into a fixed-size representation that summarizes the entire input text, particularly for downstream tasks.

Specifically, in the context of BERT and SBERT:

Aggregates token embeddings: BERT processes text at the token level, generating a high-dimensional vector for each word in the input sentence. The pooling layer takes these individual token embeddings and combines them into a single vector representation for the entire sentence.

Generates sentence embeddings: This process allows SBERT to create sentence embeddings, which are fixed-size representations that capture the meaning of a sentence.

Enables downstream tasks: These sentence embeddings are then used for various tasks, such as semantic search, clustering, and text classification.

Common pooling strategies employed in BERT/SBERT include:

Mean Pooling: This involves averaging the embeddings of all tokens in the sentence.

Max Pooling: This strategy selects the maximum value across each dimension of the token embeddings.

CLS Pooling: This method utilizes the embedding of the special [CLS] token (which is added at the beginning of each input sequence) as the sentence embedding. The [CLS] token is trained to represent the entire sequence.

Why is pooling important?

Dimensionality Reduction: Pooling significantly reduces the dimensionality of the representation, making it more computationally efficient and easier to handle for downstream tasks.

Fixed-size representation: Regardless of the length of the input sentence, pooling produces a fixed-size sentence embedding, allowing for consistent input to subsequent layers or models.

Capturing overall meaning: While BERT captures rich contextual information at the token level, pooling enables the aggregation of this information to represent the holistic meaning of the sentence.

In essence, the pooling layer acts as a bridge between the detailed token-level understanding of BERT and the need for a single, concise representation of the entire sentence for various applications.

Details:

1. BERT's Output:

BERT processes input text at the token level, generating a sequence of token embeddings (vectors).

For example, a sentence like *"This is a sentence"* would result in a sequence of embeddings, one for each token: *"This", "is", "a", "sentence"*.

These token embeddings capture contextual information about each word within the sentence.

2. The Need for Pooling:

Many downstream tasks, like semantic similarity search or sentence classification, require a single, fixed-size vector representation for each sentence.

The pooling layer addresses this by aggregating the token embeddings into a single vector, effectively summarizing the sentence's meaning.

3. Common Pooling Methods:

Mean Pooling:

This is a simple and common method where the average of all token embeddings is calculated.

Max Pooling:

This method selects the maximum value for each dimension across all token embeddings.

CLS Pooling:

In BERT, the "[CLS]" token is specifically designed to represent the entire input sequence. The embedding of this token can be used as a sentence embedding, especially in classification tasks.

SBERT's Approach:

SBERT often uses mean pooling as a default, but other methods like max pooling or CLS pooling can also be used.

4. Benefits of Pooling:

Fixed-size representation: Enables comparison of sentences of different lengths.

Semantic understanding: The pooling layer aims to capture the overall meaning of the sentence, not just individual word meanings.

Efficiency: Fixed-size vectors are computationally more efficient for similarity search and other tasks.

In essence, the pooling layer acts as a bridge between BERT's token-level representation and the fixed-size sentence representations needed for various NLP applications.