Bidirectional Encoder Representation from Transformers(BERT)

Bert is an embedding model which is used for the creation of dense embeddings from the raw text inputted into the model and in contrast to the traditional models where text is looked into in a sequential manner i.e in one direction this model tries to read and gain understanding of the sentences from opposite directions to understand the semantic meaning and capture the relationships between the words in that sentence.

Bert has been categorized as an unique case due to the pre-training stratagies involved in that :

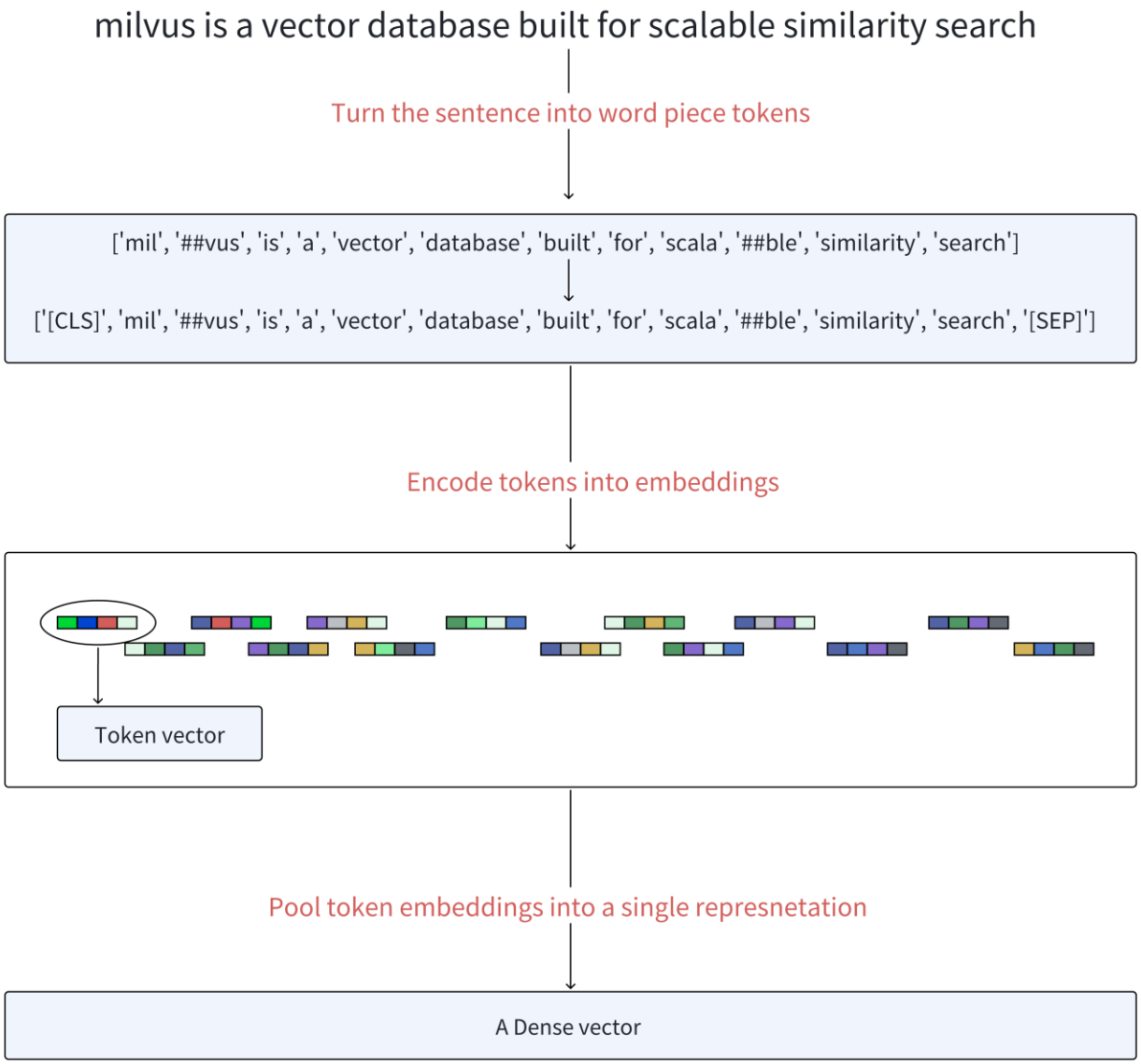
MLM(Masked Language Modelling): Here the input tokens are randomly masked and then the model is trained to identify these tokens and this technique is mostly preferred for training models to excel in the task of next word generation and unlike in traditional models bert fills these masked tokens by understanding the full context of the sentence as it reads the sentence from two sides and connects the semantic words/tokens to be filled in these gaps.

NSP(Next Sentence Prediction): Here the model is trained such as to predict what sentence comes next after the initial sentence is formed so that the context makes sense as normally found in articles/textbooks.

Since BERT also uses the transformers architecture it is important to note that the self attention mechanism is used by each encoder layer to weigh the importance of other words in sentences to understand how different words can be used to get the same or different meaning in various contexts as required.

Note that most of the models always use positional encoding to enable for the tokens which are formed from the text to be formed to have their positions jotted down to not lose the nuanced sequential meaning of a sentence.

The following is the process which is carried out in BERT when it is prompted:



Tokenization: the passed text is converted into tokens which can be either full words or partial words. The [CLS] token is prepended to the input for sentence-level tasks, and [SEP] tokens are inserted to separate sentences and indicate the end.

Embedding: Now that the tokens are available it is converted ton embeddings which are a higher dimensional vectors utilized for further purposes this conversion can take place with the help of various techniques but for bert we utilize embedding matrix like word2vec and positional embeddings are added to these token embeddings to maintain the sequence within the sentence and segment

embeddings are also added so to differentiate these sentences and also maintain their contextual meaning.

Encoding: The generated vectors are now passed through encoders containing the feed forward neural network layers each employing self attention mechanism on the tokens thus deepening the understanding of their relationships based on the context provided by other tokens present in sequence.

Output layer: This is the final layer which outputs a sequence of embeddings containing all the information which has been studied and these embeddings are dense embeddings. Typically, for sentence-level tasks, the embeddings of the [CLS] token are the aggregate representation of the entire input. Embeddings of individual tokens are utilized for fine-grained tasks or combined via operations like max or sum pooling to form a singular dense representation.

The generated dense embeddings capture meaning of each and every single word present within the sentence contained in the dense embeddings and their interrelations and this approach has proven immensely successful in various NLP tasks by improving the quality of the responses generated.