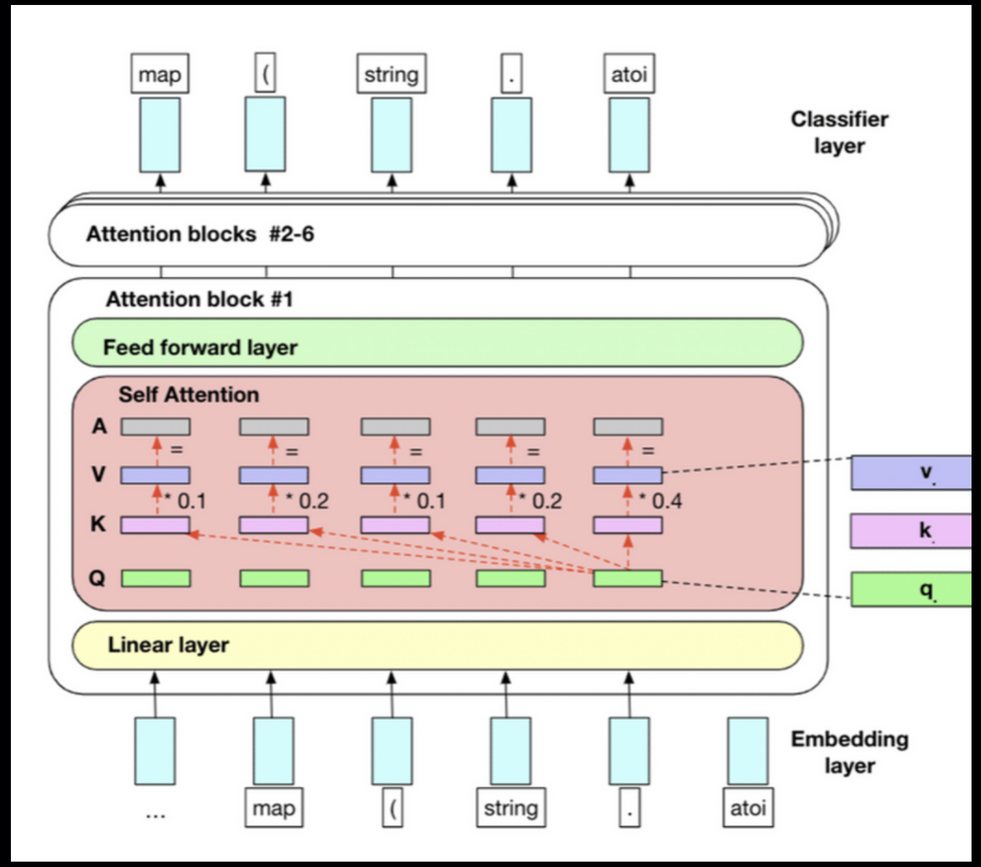
- # Why use Transformers :

Transformers are a type of neural network architecture that has been shown to be highly effective in natural language processing tasks such as text summarization. Unlike traditional recurrent neural networks, transformers can process entire sequences of text simultaneously, allowing them to capture complex dependencies between different parts of the input.

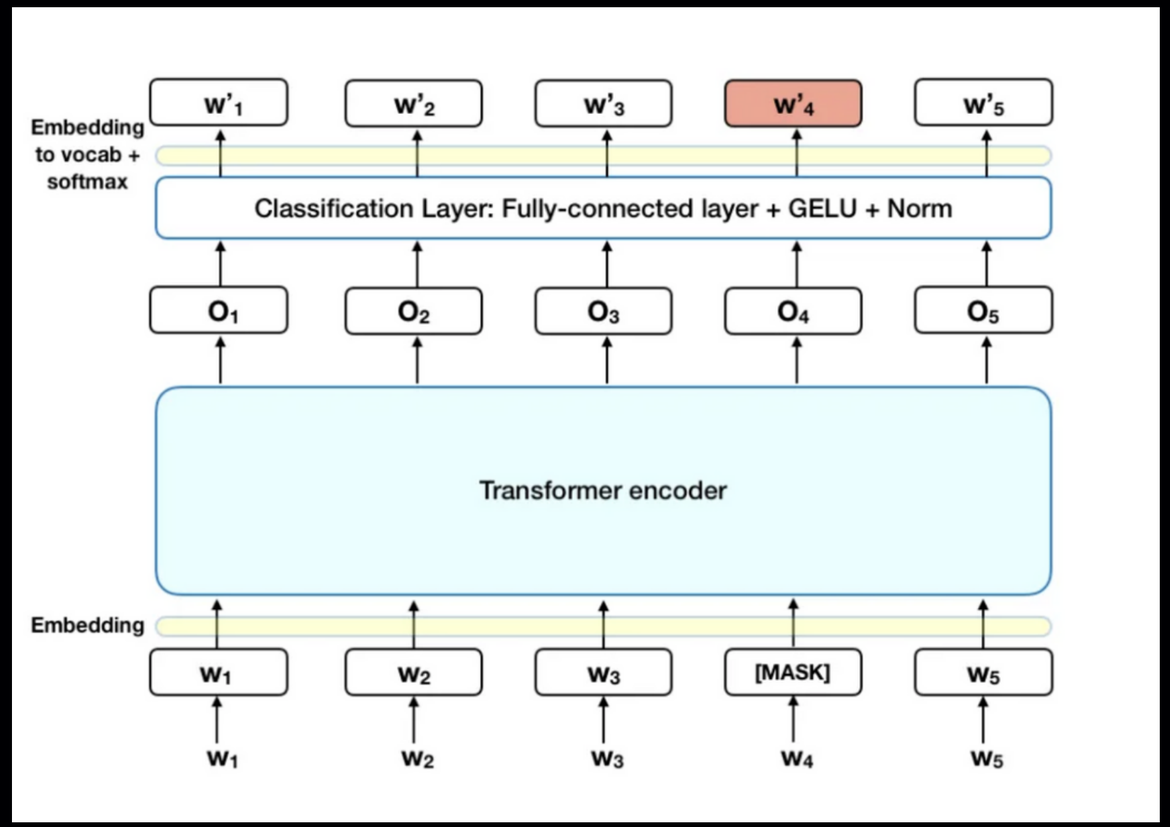
In particular, transformers use a self-attention mechanism that allows them to focus on different parts of the input during different stages of processing. This allows them to learn more effective representations of the input, leading to better performance in text summarization tasks. Additionally, transformers can be pre-trained on large corpora of text using unsupervised learning, which can improve their performance on downstream tasks like text summarization.

Some of the most popular transformer models for text summarization are:

1. **BERT**(Bidirectional Encoder Representations from Transformers)
2. **GPT**(Generative Pre-trained Transformer)
3. **T5**(Text-to-Text Transfer Transformer)

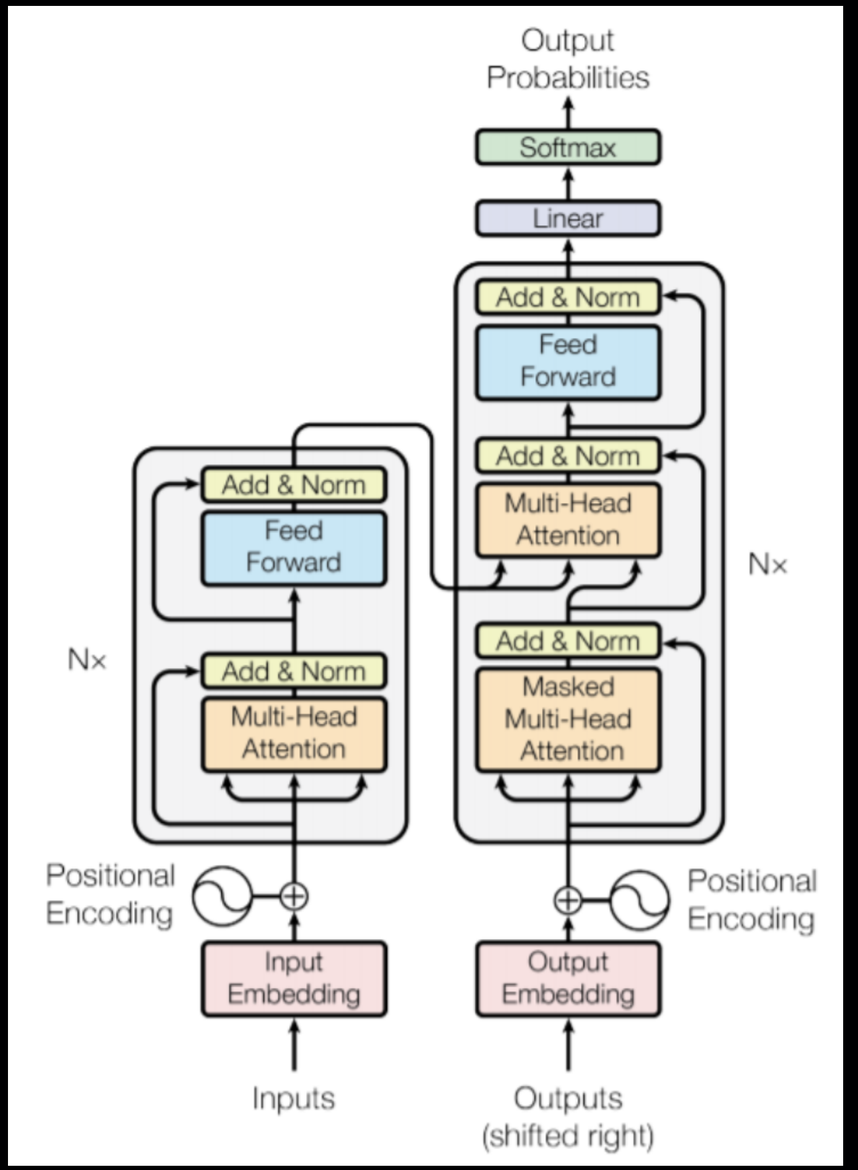


GPT transformer architecture

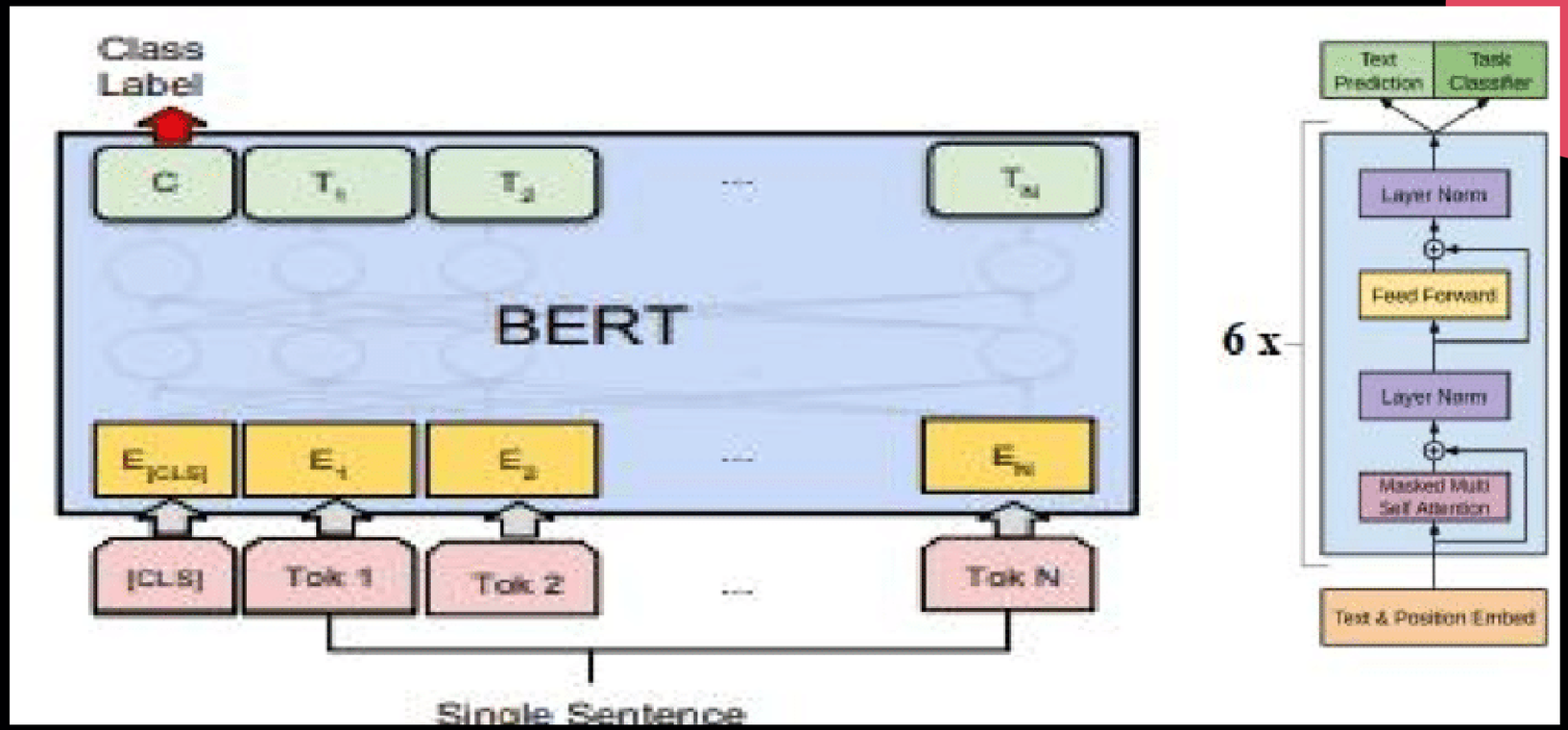


BERT transformer architecture

T5 transformer architecture



Text classification model used: **DistilBERT**



DistilBERT transformer architecture

All of these transformer models share a similar basic architecture, which consists of an encoder and a decoder. The encoder processes the input sequence and generates a representation of the input, while the decoder uses the encoder representation to generate the output summary. However, each of these models differs in the details of their architecture, pre-training objectives, and fine-tuning procedures, which can lead to different performance on text summarization tasks.

- Why Transformer above RNN/CNN/LSTM ?

Transformers are preferred over traditional RNNs, CNNs, and LSTMs for natural language processing tasks like text summarization due to their ability to handle long-range dependencies, unsupervised pre-training, and parallel processing of entire input sequences. These advantages make transformers better suited for capturing complex relationships between different parts of the input and improving performance on downstream tasks. In contrast, RNNs and LSTMs can struggle with long-range dependencies and require more labeled training data to achieve high performance. Overall, transformers are a powerful and effective tool for NLP tasks like text summarization.

- ROUGE Score :

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is a set of evaluation metrics commonly used in natural language processing (NLP) and text summarization tasks to measure the quality of the generated summary with respect to the reference summary or the ground truth.

There are several variants of the ROUGE score, including ROUGE-1 (unigram overlap), ROUGE-2 (bigram overlap), ROUGE-L (longest common subsequence-based overlap), and ROUGE-S (skip-bigram overlap), among others. Each variant of ROUGE score measures a different aspect of summary quality and provides a different perspective on the effectiveness of the summarization algorithm.

```
Reference Tokens: ['John','really','loves','data','science'](n_r=5)
Candidate Tokens: ['John','loves','data','science'] (n_can=4)
Captured Tokens: ['John','loves','data','science'] (n_cap=4)

Rouge-1 Recall = n_cap/n_r = 4/5
Rouge-1 Precision = n_cap/n_can = 4/4
```

- Classification Metrices :

1. **Accuracy**: Accuracy measures the percentage of correctly classified instances out of the total number of instances. In text classification, accuracy represents the proportion of correctly classified documents or text snippets out of the total number of documents.
2. **Precision**: Precision measures the proportion of true positive instances (correctly classified instances) out of the total number of instances that were classified as positive. In text classification, precision represents the proportion of correctly classified positive instances (e.g., documents or text snippets that belong to a specific class) out of the total number of instances that were classified as positive.
3. **Recall**: Recall measures the proportion of true positive instances out of the total number of instances that actually belong to the positive class. In text classification, recall represents the proportion of correctly classified positive instances out of the total number of instances that actually belong to the positive class.
4. **F1 score**: The F1 score is a weighted average of precision and recall, calculated as 2 * (precision * recall) / (precision + recall). The F1 score provides a balance between precision and recall and is often used as an overall evaluation metric in text classification tasks.