

Artificial Intelligence

Lecture09 – BERT



Contenido

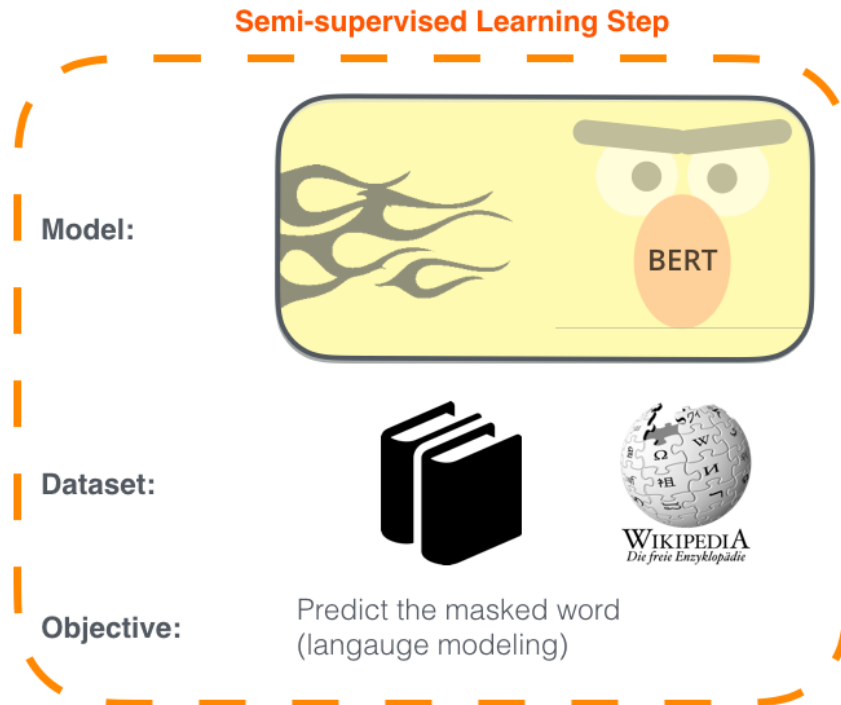
1. Learning problems
2. Model architecture
3. Pre-training
4. Applications
5. Hands-on



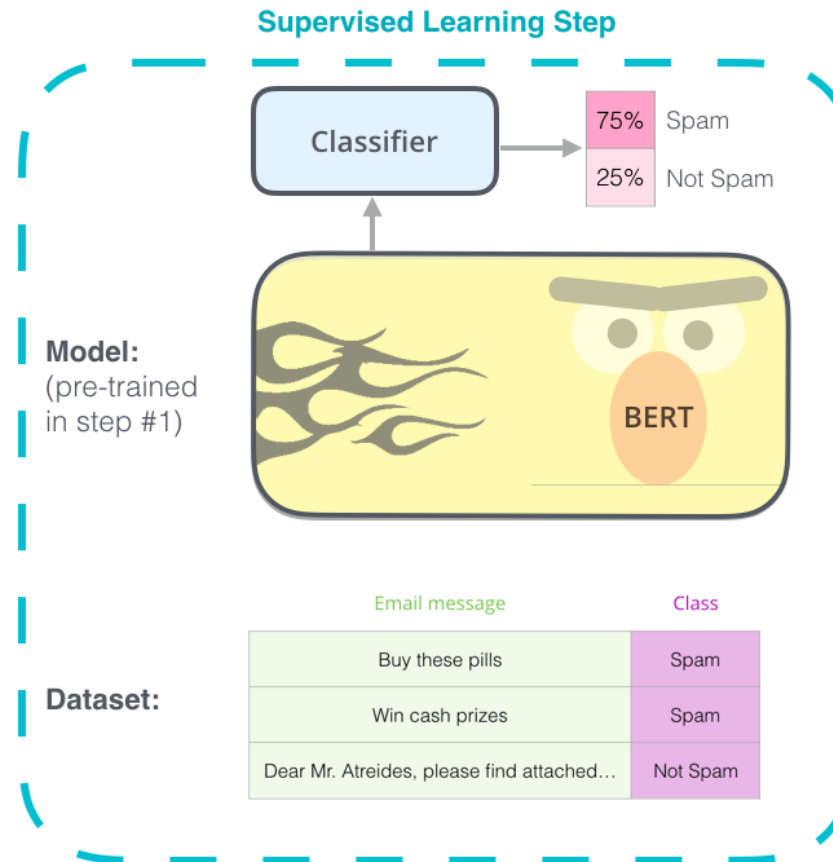
Self supervised learning

1 - **Semi-supervised** training on large amounts of text (books, wikipedia..etc).

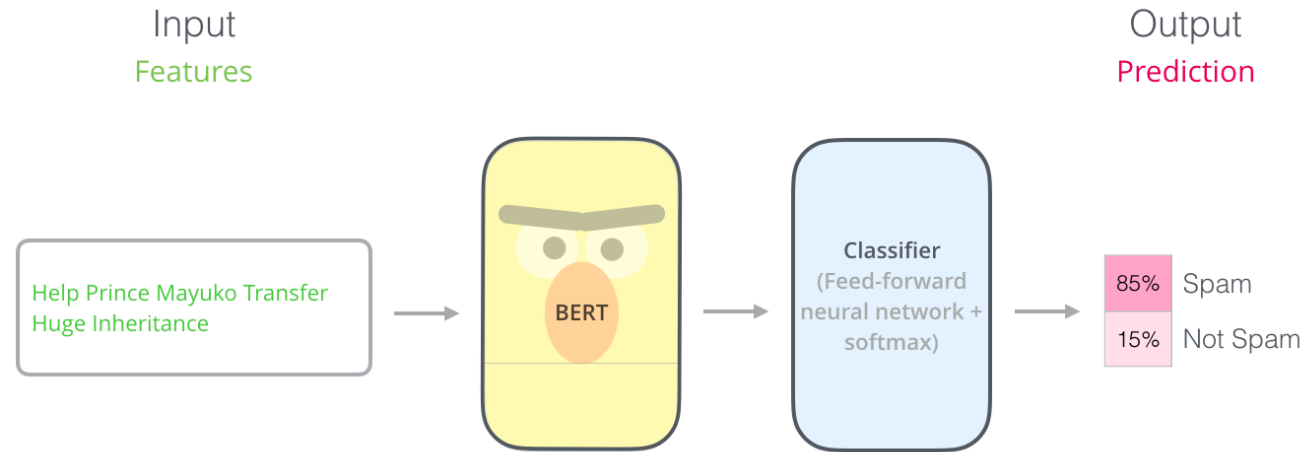
The model is trained on a certain task that enables it to grasp patterns in language. By the end of the training process, BERT has language-processing abilities capable of empowering many models we later need to build and train in a supervised way.



2 - **Supervised** training on a specific task with a labeled dataset.



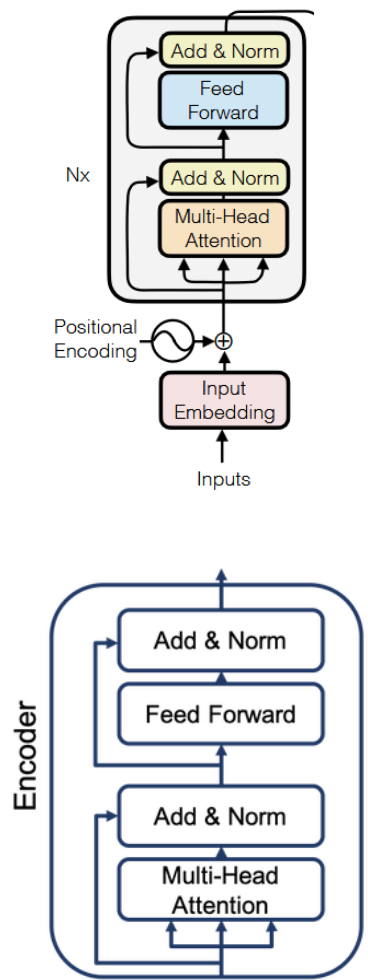
Transfer learning



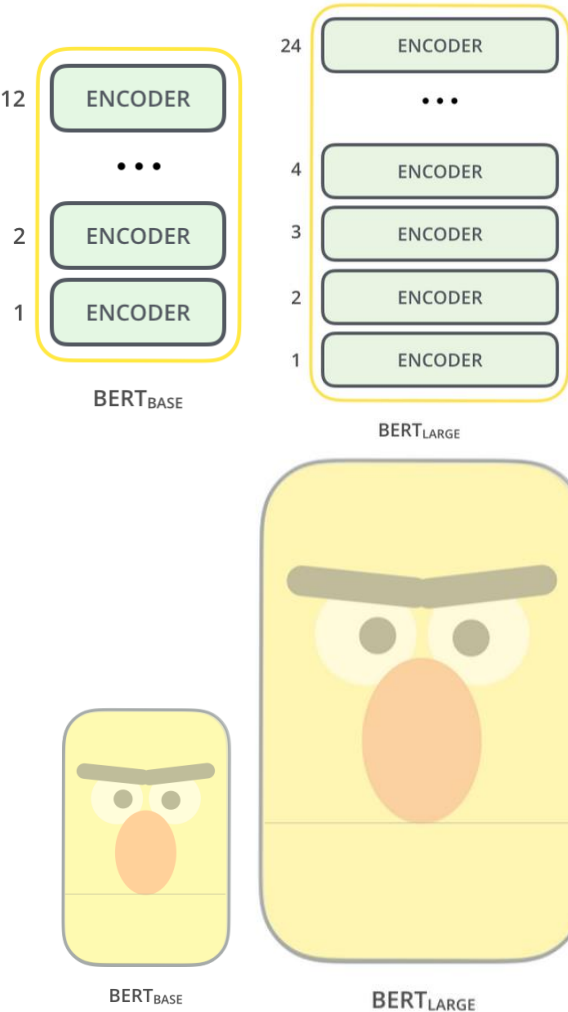
Email message	Class
Buy these pills	Spam
Win cash prizes	Spam
Dear Mr. Atreides, please find attached...	Not Spam



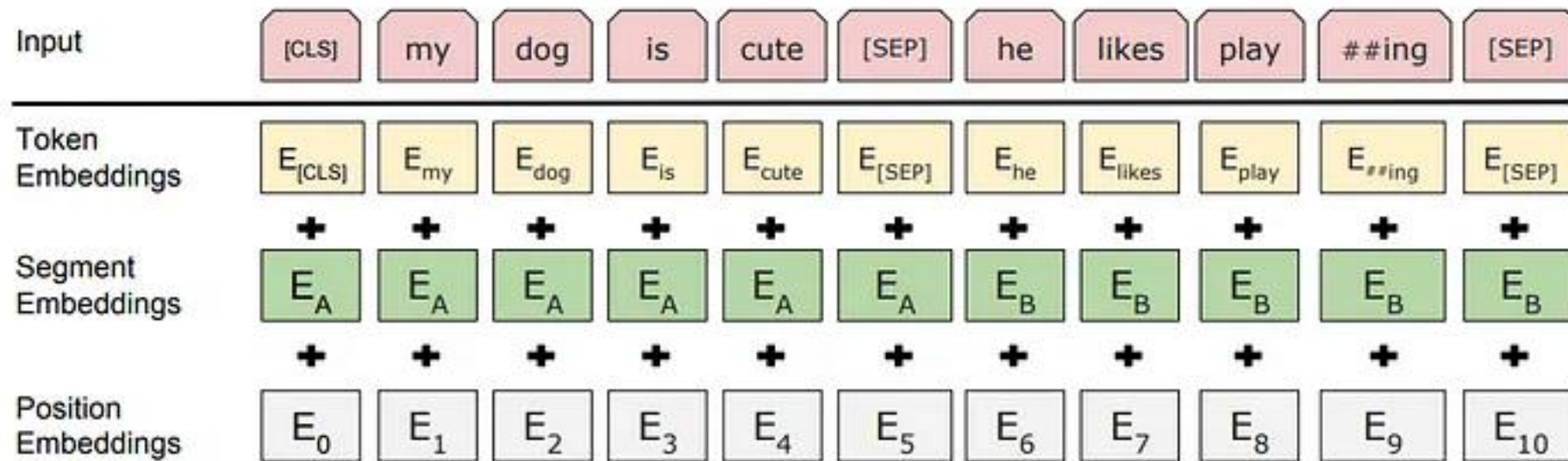
Model architecture



<code>bert-base-uncased</code>	12-layer, 768-hidden, 12-heads, 110M parameters. Trained on lower-cased English text.
<code>bert-large-uncased</code>	24-layer, 1024-hidden, 16-heads, 340M parameters. Trained on lower-cased English text.
<code>bert-base-cased</code>	12-layer, 768-hidden, 12-heads, 110M parameters. Trained on cased English text.
<code>bert-large-cased</code>	24-layer, 1024-hidden, 16-heads, 340M parameters. Trained on cased English text.
<code>bert-base-multilingual-uncased</code>	(Original, not recommended) 12-layer, 768-hidden, 12-heads, 110M parameters. Trained on lower-cased text in the top 102 languages with the largest Wikipedias (see details).
<code>bert-base-multilingual-cased</code>	(New, recommended) 12-layer, 768-hidden, 12-heads, 110M parameters. Trained on cased text in the top 104 languages with the largest Wikipedias (see details).



Model inputs



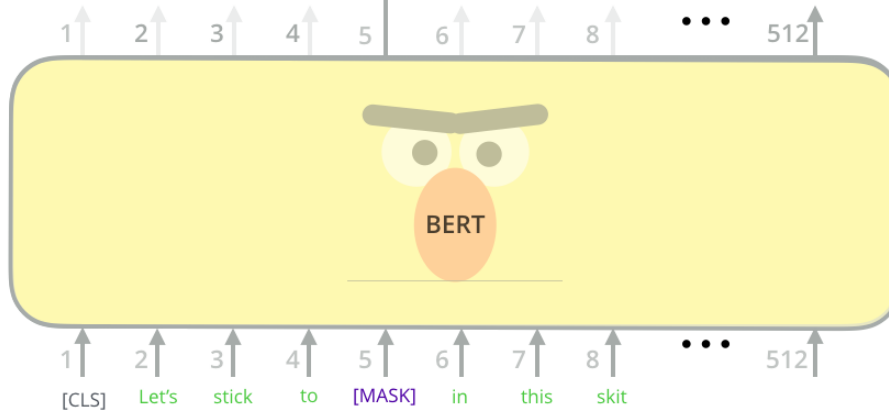
Pre-training: Masked Language Model

Use the output of the masked word's position to predict the masked word

Possible classes:
All English words

0.1%	Aardvark
...	...
10%	Improvisation
...	...
0%	Zyzyva

FFNN + Softmax



Randomly mask
15% of tokens

Input

[CLS] Let's stick to improvisation in this skit

15%

80%

10%

10%

my dog is [MASK]

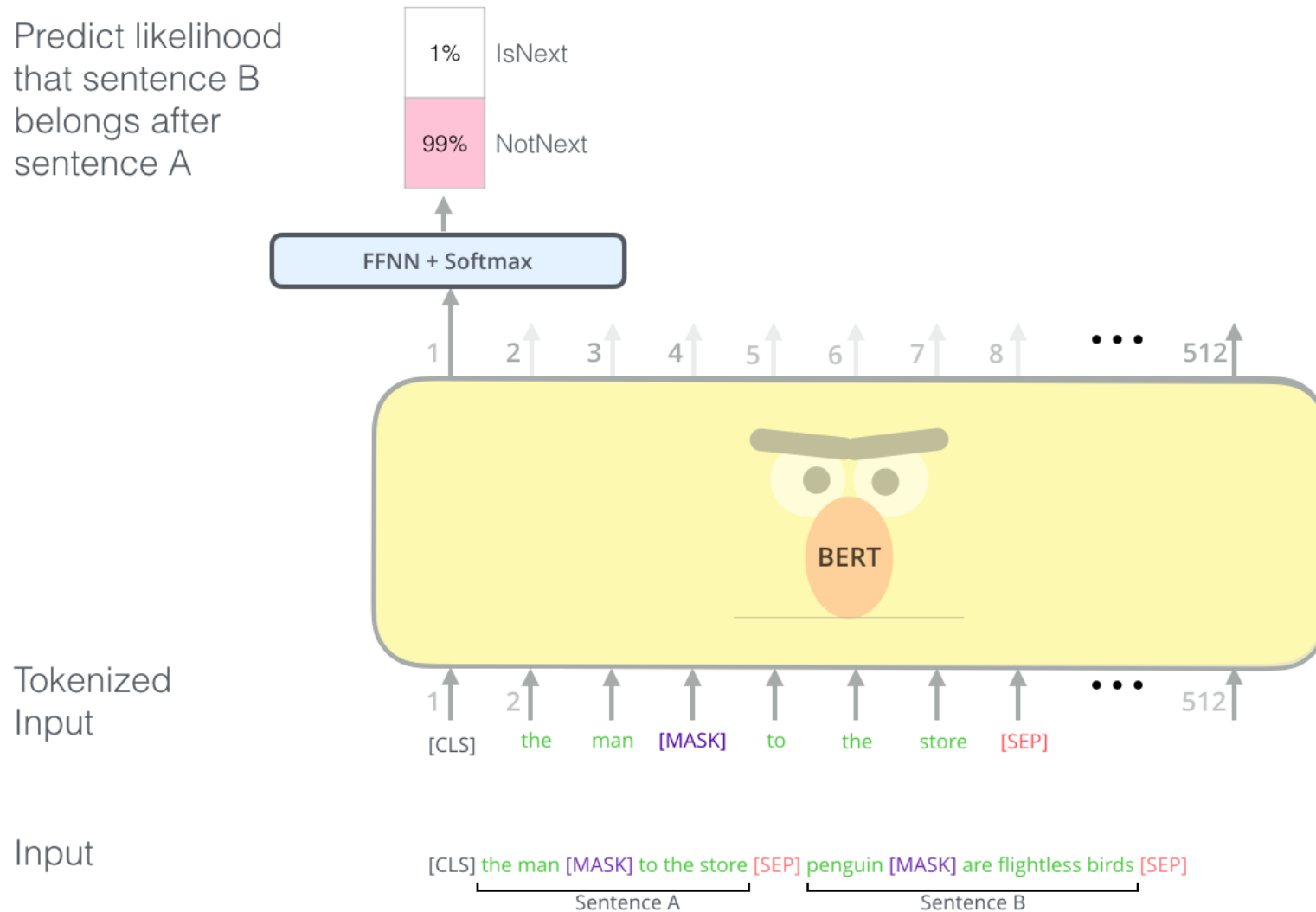
my dog is apple

my dog is hairy



Pre-training: Two sentences task

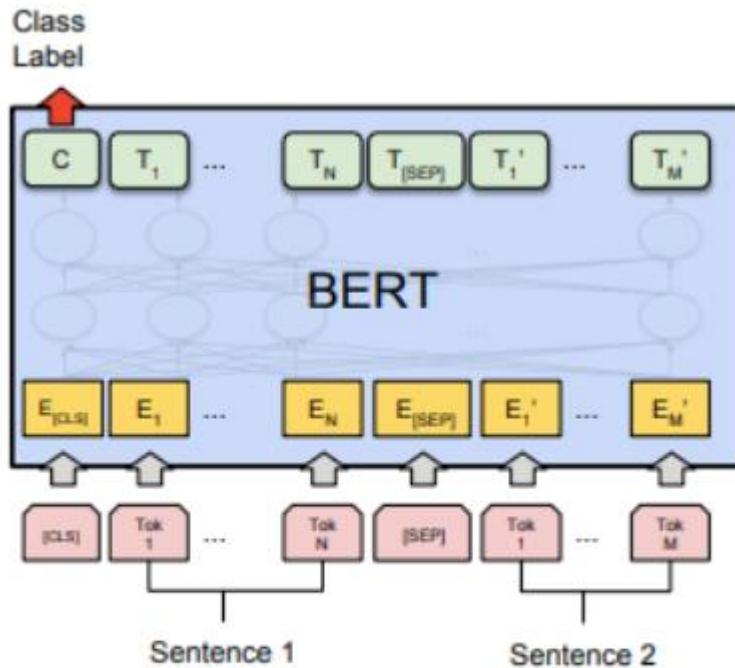
Predict likelihood
that sentence B
belongs after
sentence A



Input	[CLS] the man went to the [MASK] store [SEP] he bought a gallon [MASK] milk [SEP]
Label	IsNext
Input	[CLS] the man went to the [MASK] store [SEP] penguin [MASK] are flight ##less birds [SEP]
Label	NotNext



BERT Specific tasks



(a) Sentence Pair Classification Tasks:
MNLI, QQP, QNLI, STS-B, MRPC,
RTE, SWAG

📌 **Objective:** Determine the relationship between two sentences.

✅ **Input:**

- Two sentences are concatenated with a [SEP] token in between.
- [CLS] is added at the beginning.
- Each token gets embeddings (**word, segment, positional**).

✅ **BERT Processing:**

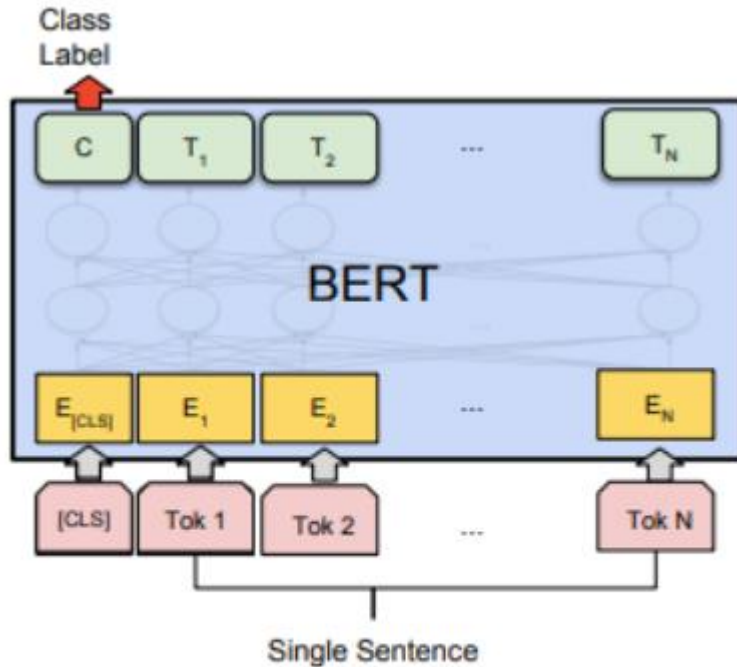
- Applies **self-attention** over both sentences.
- [CLS] token is used to represent the **entire input**.

✅ **Output:**

- A **classifier** (fully connected layer + softmax) is applied to the [CLS] token to predict the **relationship** between sentences (e.g., entailment, contradiction).



BERT Specific tasks



(b) Single Sentence Classification Tasks:
SST-2, CoLA

📌 **Objective:** Classify the **entire sentence** into a category (e.g., sentiment analysis).

✅ **Input:**

- A single sentence with a [CLS] token at the start.

✅ **BERT Processing:**

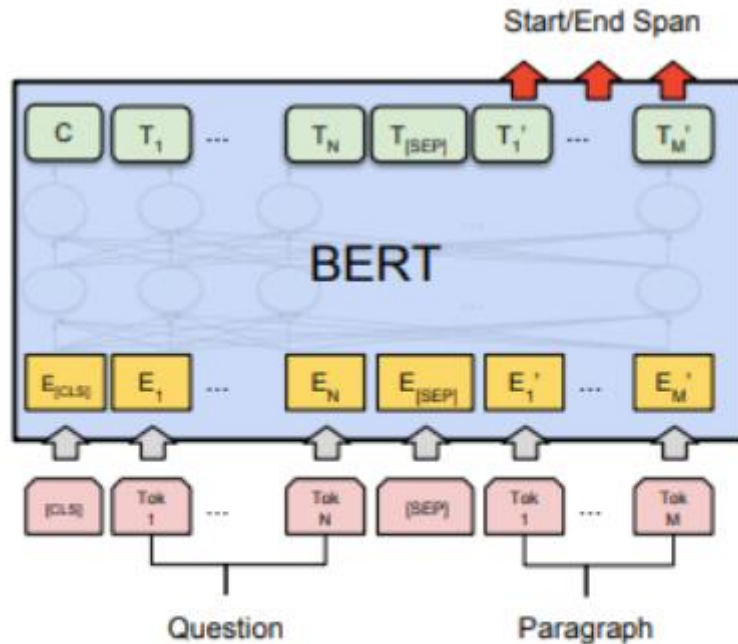
- Self-attention processes the sentence.

✅ **Output:**

- The classifier is applied to the [CLS] token, which encodes sentence-level information.



BERT Specific tasks



(c) Question Answering Tasks:
SQuAD v1.1

📌 **Objective:** Find the **start** and **end** of the answer span in a paragraph given a question.

✅ **Input:**

- The **question** and **paragraph** are separated by [SEP].

✅ **BERT Processing:**

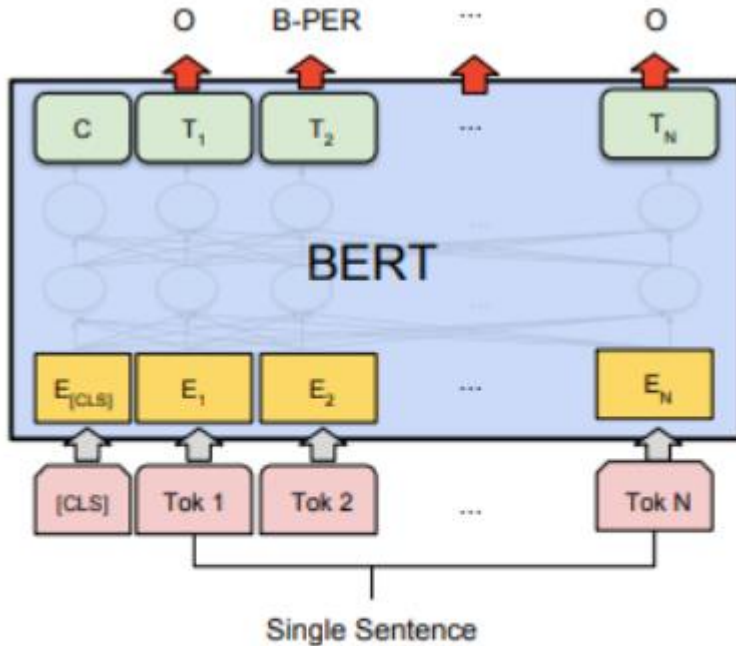
- Self-attention captures relationships between the **question** and the **paragraph**.

✅ **Output:**

- Two classifiers predict:
 1. **Start position** of the answer.
 2. **End position** of the answer.



BERT Specific tasks



(d) Single Sentence Tagging Tasks:
CoNLL-2003 NER

- 📌 **Objective:** Assign a **label** to each token in the sentence (e.g., Person, Organization, Location).
- ✅ **Input:**
 - A single sentence with a [CLS] token
- ✅ **BERT Processing:**
 - Self-attention captures context for each Word
- ✅ **Output:**
 - Instead of using the [CLS] token, **each token** has its **own classification**.
 - Labels are assigned to **each token** (e.g., "B-PER" for "beginning of a person's name").



Suggested readings

<https://huggingface.co/blog/bert-101>

<https://huggingface.co/blog/pretraining-bert>

