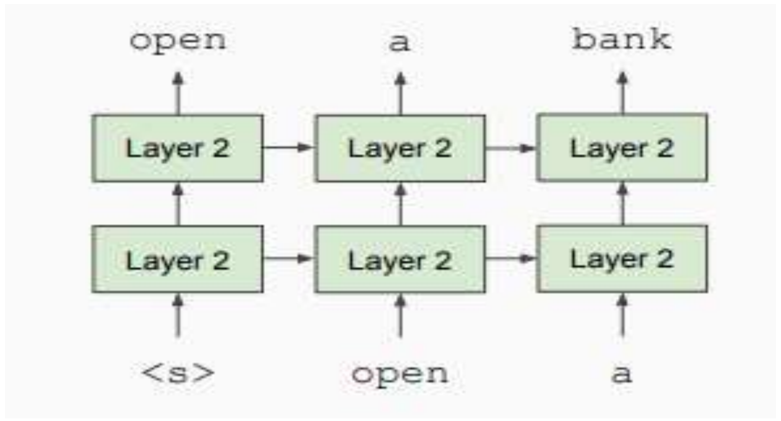
Introduction

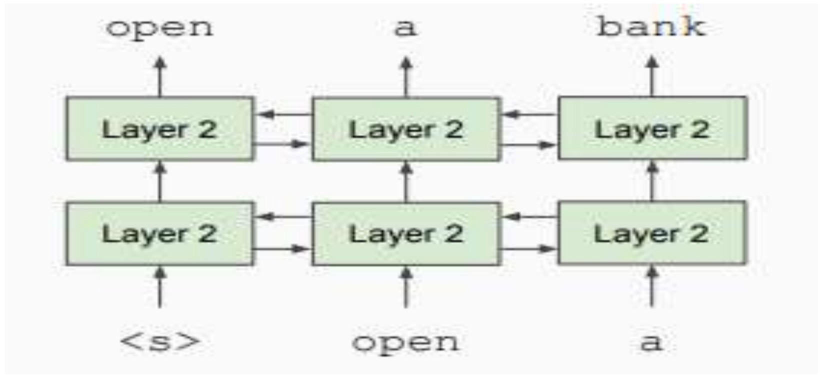
BERT was created and published in 2018 by Jacob Devlin and his colleagues from Google . It stands for **Bidirectional Encoder Representations from Transformers**

Bert is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of NLP tasks.”

Why Bi-directional Model?



Unidirectional context Build representation incrementally

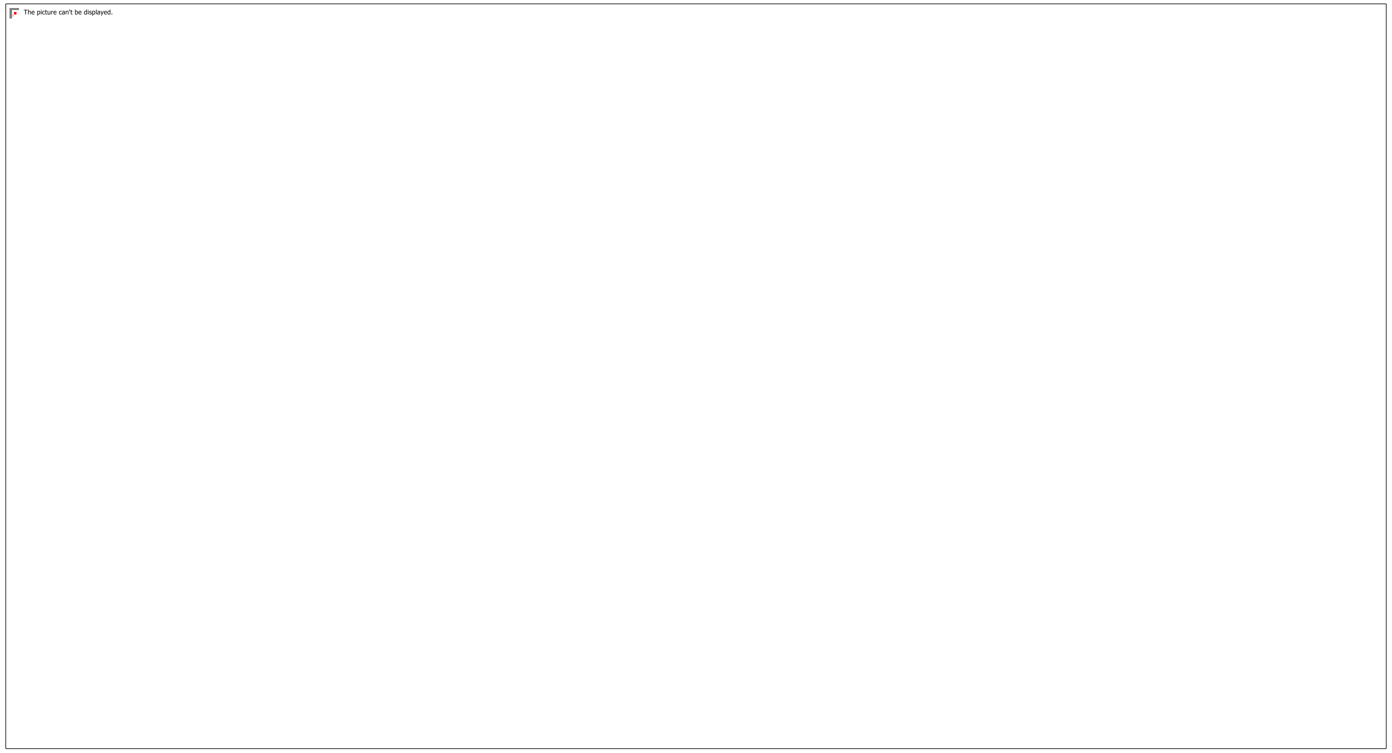


While in Bidirectional context words can “see themselves”

Architecture

The BERT architecture builds on top of Transformer. We currently have two variants available:

- **BERT Base**: 12 layers (transformer blocks), 12 attention heads, and 110 million parameters
- **BERT Large**: 24 layers (transformer blocks), 16 attention heads and, 340 million parameters



Text Preprocessing

The picture can't be displayed.

Text Preprocessing

The developers behind BERT have added a specific set of rules to represent the input text for the model. Many of these are creative design choices that make the model even better.

For starters, every input embedding is a combination of 3 embeddings:

1. **Position Embeddings**: BERT learns and uses positional embeddings to express the position of words in a sentence. These are added to overcome the limitation of Transformer which, unlike an RNN, is not able to capture "sequence" or "order" information
2. **Segment Embeddings**: BERT can also take sentence pairs as inputs for tasks (Question-Answering). That's why it learns a unique embedding for the first and the second sentences to help the model distinguish between them. In the above example, all the tokens marked as EA belong to sentence A (and similarly for EB)
3. **Token Embeddings**: These are the embeddings learned for the specific token from the WordPiece token vocabulary

Pre-training Tasks

BERT is pre-trained on two NLP tasks:

- ❑ Masked Language Modeling
- ❑ Next Sentence Prediction

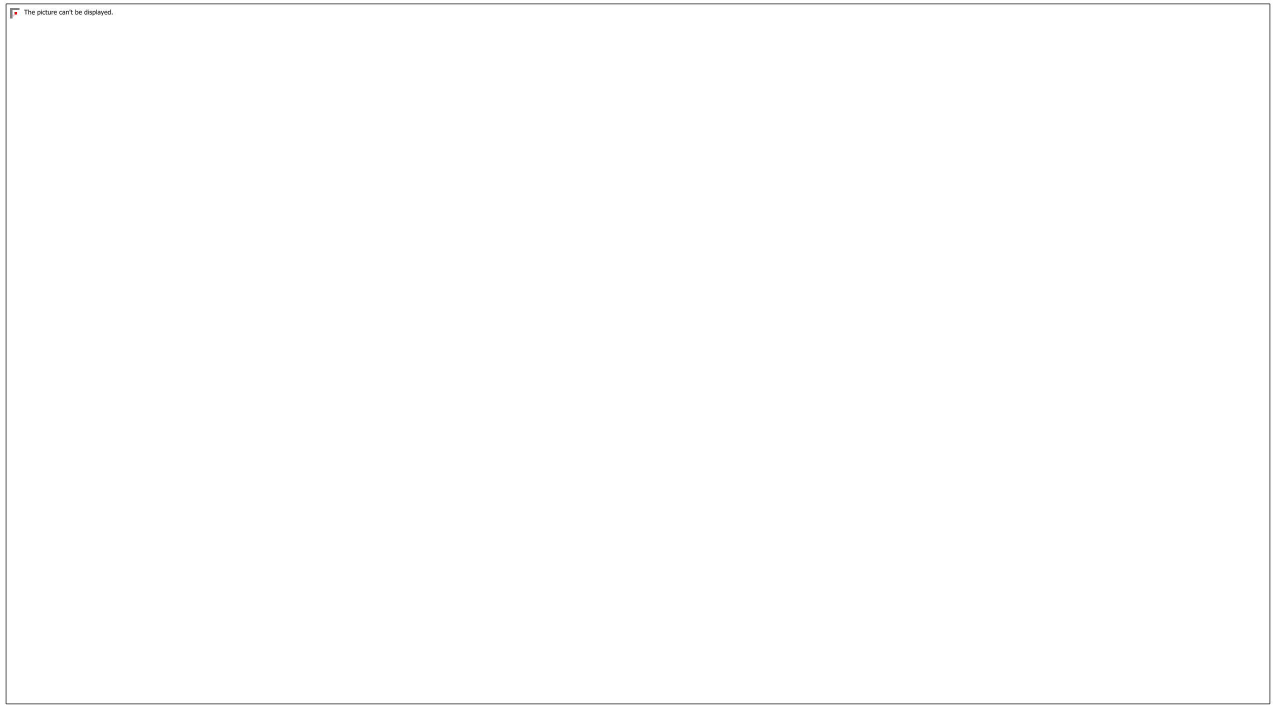
The objective of **Masked Language Model (MLM)** training is to hide a word in a sentence and then have the program predict what word has been hidden (masked) based on the hidden word's context.

The objective of **Next Sentence Prediction** training is to have the program predict whether two given sentences have a logical, sequential connection or whether their relationship is simply random.

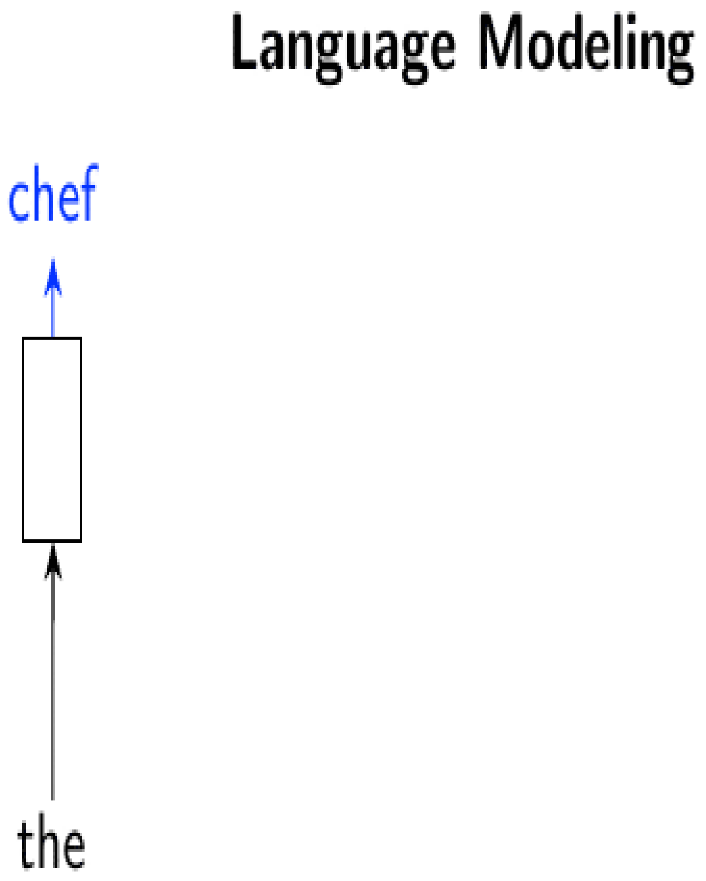
Masked Language Modeling

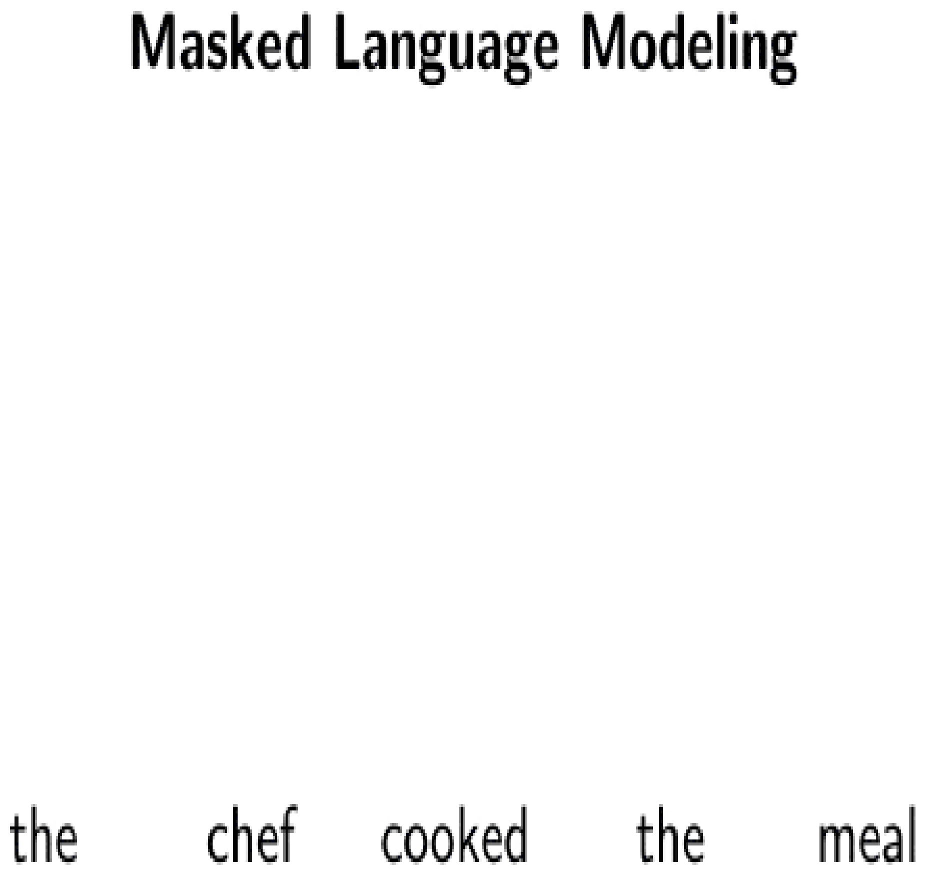
The **Bert** network effectively captures information from both the right and left context of a token from the first layer itself and all the way through to the last layer.

Traditionally, we had language models either trained to predict the next word in a sentence (right-to-left context used in GPT) or language models that were trained on a left-to-right context. This made our models susceptible to errors due to loss in information.



Masked Language Modeling





Masked Language Modeling

The authors of BERT also include some caveats to further improve this technique:

- ❑ To prevent the model from focusing too much on a particular position or tokens that are masked, the researchers randomly masked 15% of the words
- ❑ The masked words were not always replaced by the masked tokens [MASK] because the [MASK] token would never appear during fine-tuning
- ❑ So, the researchers used the below technique:
  - ➢ 80% of the time the words were replaced with the masked token [MASK]
  - ➢ 10% of the time the words were replaced with random words
  - ➢ 10% of the time the words were left unchanged

Next Sentence Prediction

To learn relationships between sentences, predict whether Sentence B is actual sentence that proceeds Sentence A, or a random sentence

Sentence A = The man went to the store.
Sentence B = He bought a packet of chips.
Label = IsNextSentence

Sentence A = The man went to the store.
Sentence B = Penguins are flightless
Label = NotNextSentence

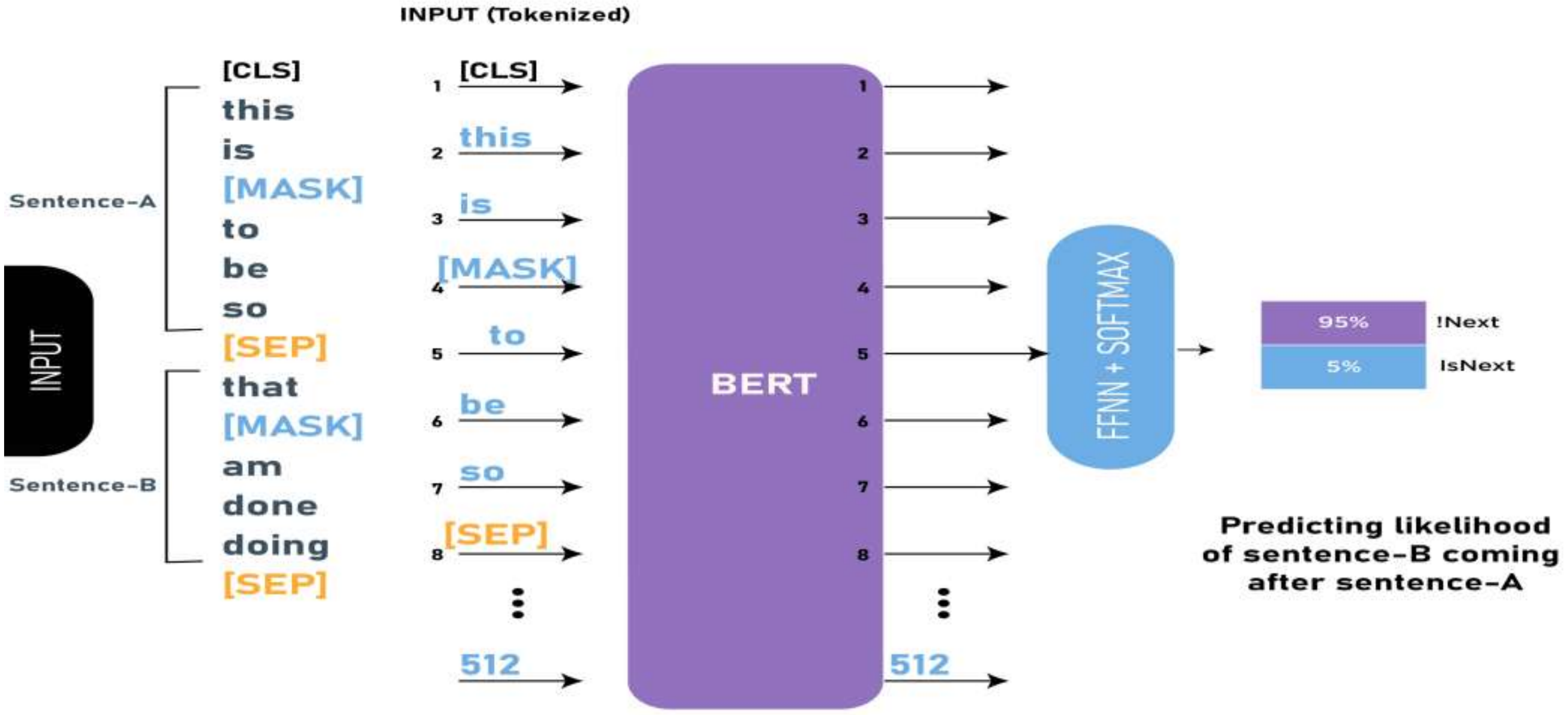
Next Sentence Prediction

Since it is a binary classification task, the data can be easily generated from any corpus by splitting it into sentence pairs. Just like MLMs, the authors have added some caveats here too. Let’s take this with an example:

Consider that we have a text dataset of 100,000 sentences. So, there will be 50,000 training examples or pairs of sentences as the training data.

- ❑ For 50% of the pairs, the second sentence would actually be the next sentence to the first sentence
- ❑ For the remaining 50% of the pairs, the second sentence would be a random sentence from the corpus
- ❑ The labels for the first case would be ***‘IsNext’*** and ***‘NotNext’*** for the second case

Next Sentence Prediction

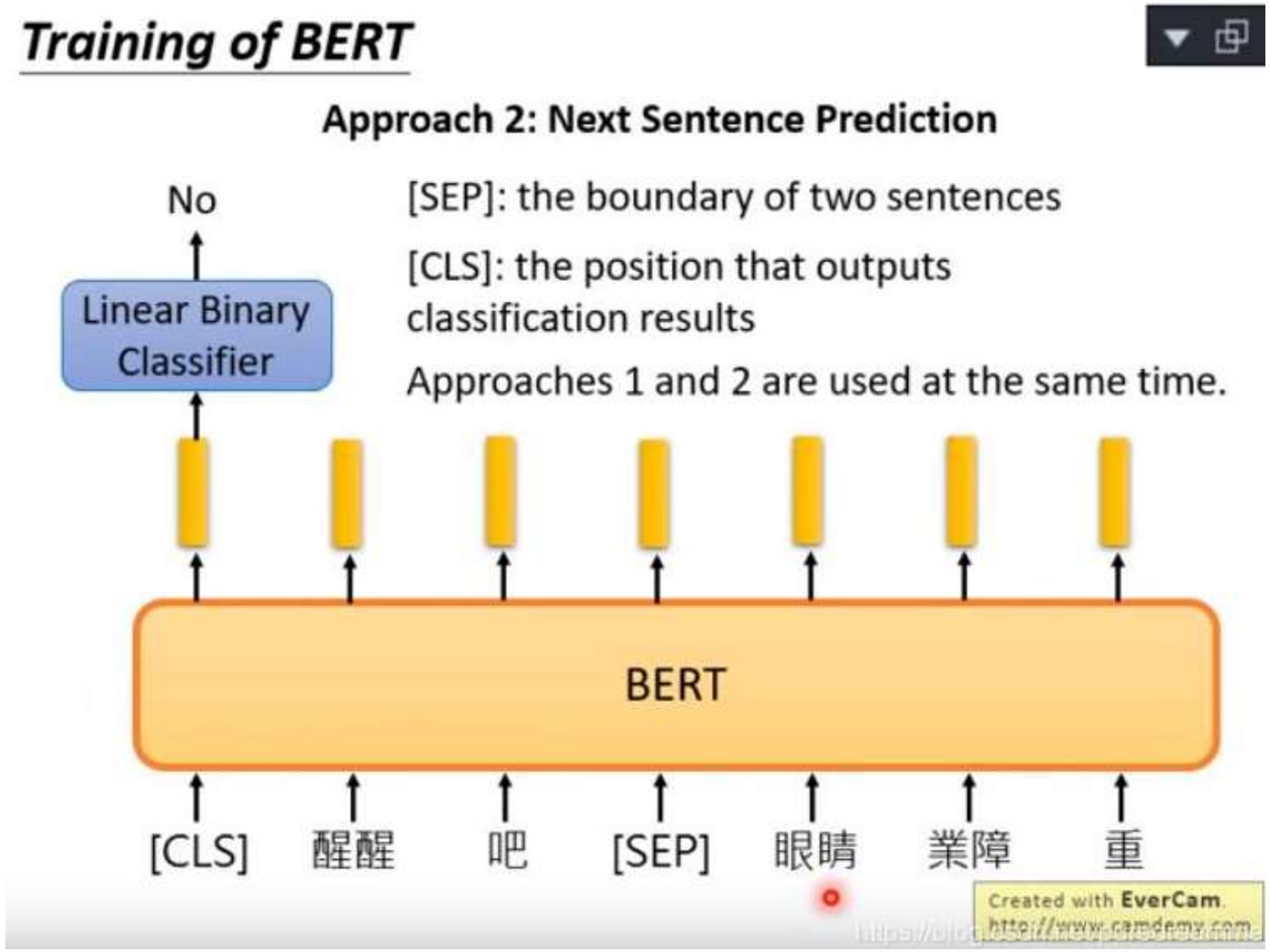


Next Sentence Prediction

To help the model distinguish between the two sentences in training, the input is processed in the following way before entering the model:

- ❑ A [CLS] token is inserted at the beginning of the first sentence and a [SEP] token is inserted at the end of each sentence.
- ❑ A sentence embedding indicating Sentence A or Sentence B is added to each token. Sentence embeddings are similar in concept to token embeddings with a vocabulary of 2.
- ❑ A positional embedding is added to each token to indicate its position in the sequence.

Next Sentence Prediction



Training of BERT

Approach 2: Next Sentence Prediction

[SEP]: the boundary of two sentences

[CLS]: the position that outputs classification results

Approaches 1 and 2 are used at the same time.

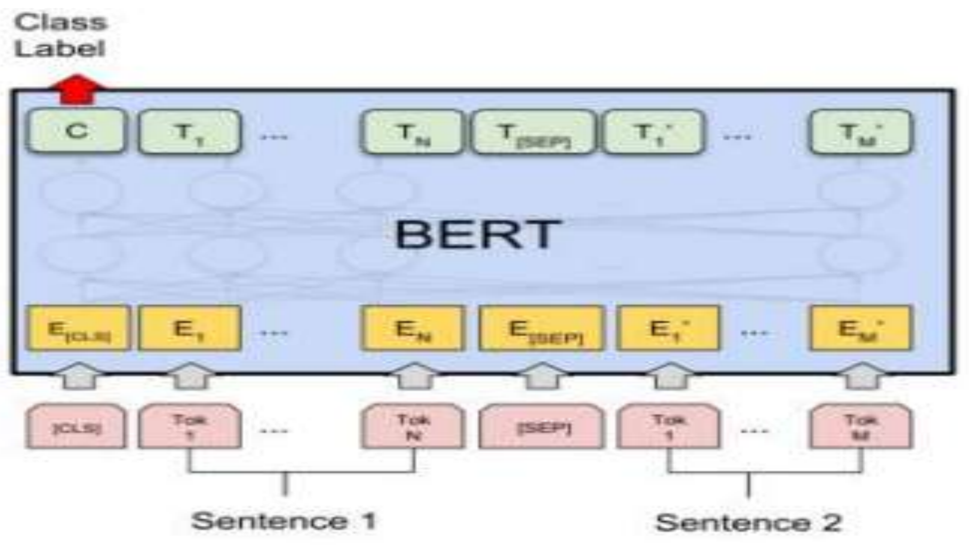
Fine-tuning with BERT

- Context vector C: Take the final hidden state corresponding to the first token in the input: [CLS].
- Transform to a probability distribution of the class labels:

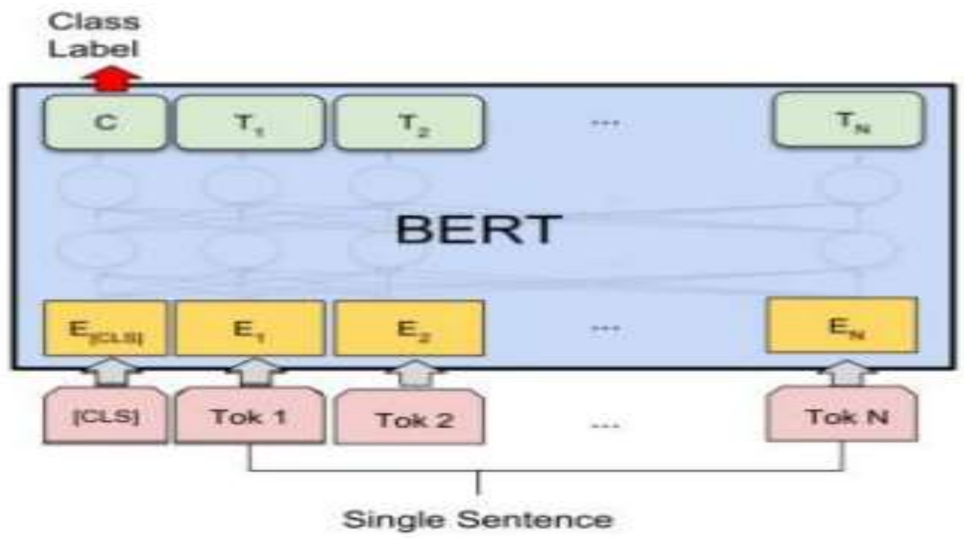
$$P = \mathrm{softmax}(CW^T)$$

- **Batch size**: 16, 32
- **Learning rate (Adam)**: 5e-5, 3e-5, 2e-5
- **Number of epochs**: 3, 4

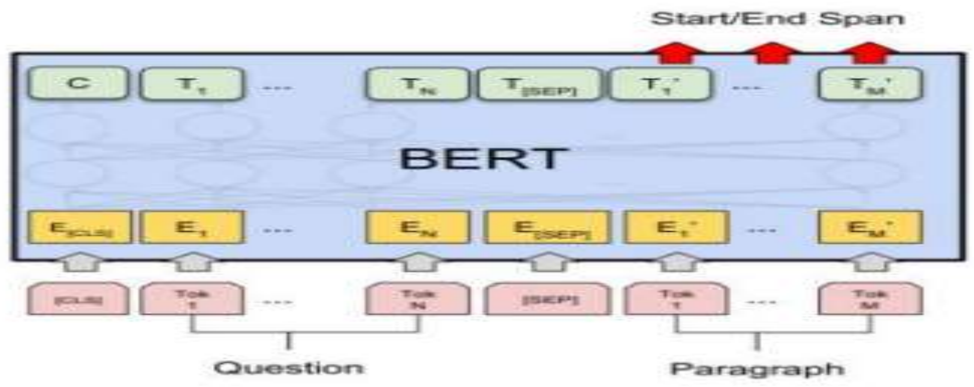
Fine-tuning with BERT



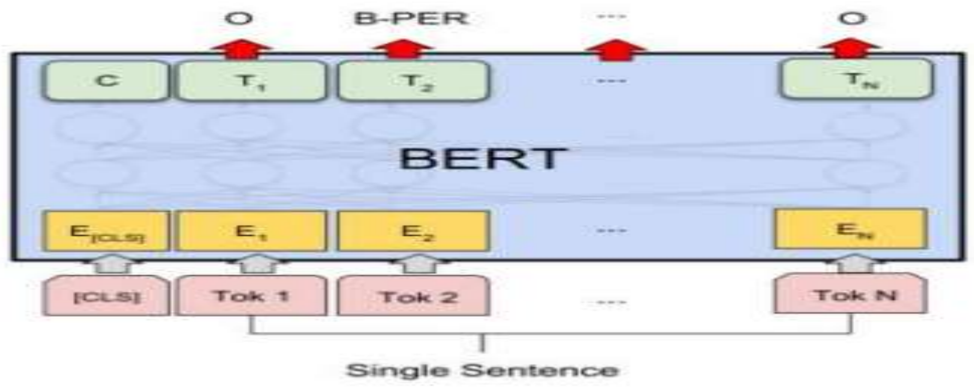
(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG



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



(c) Question Answering Tasks: SQuAD v1.1



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

BERT in a Nutshell

- Input Question:

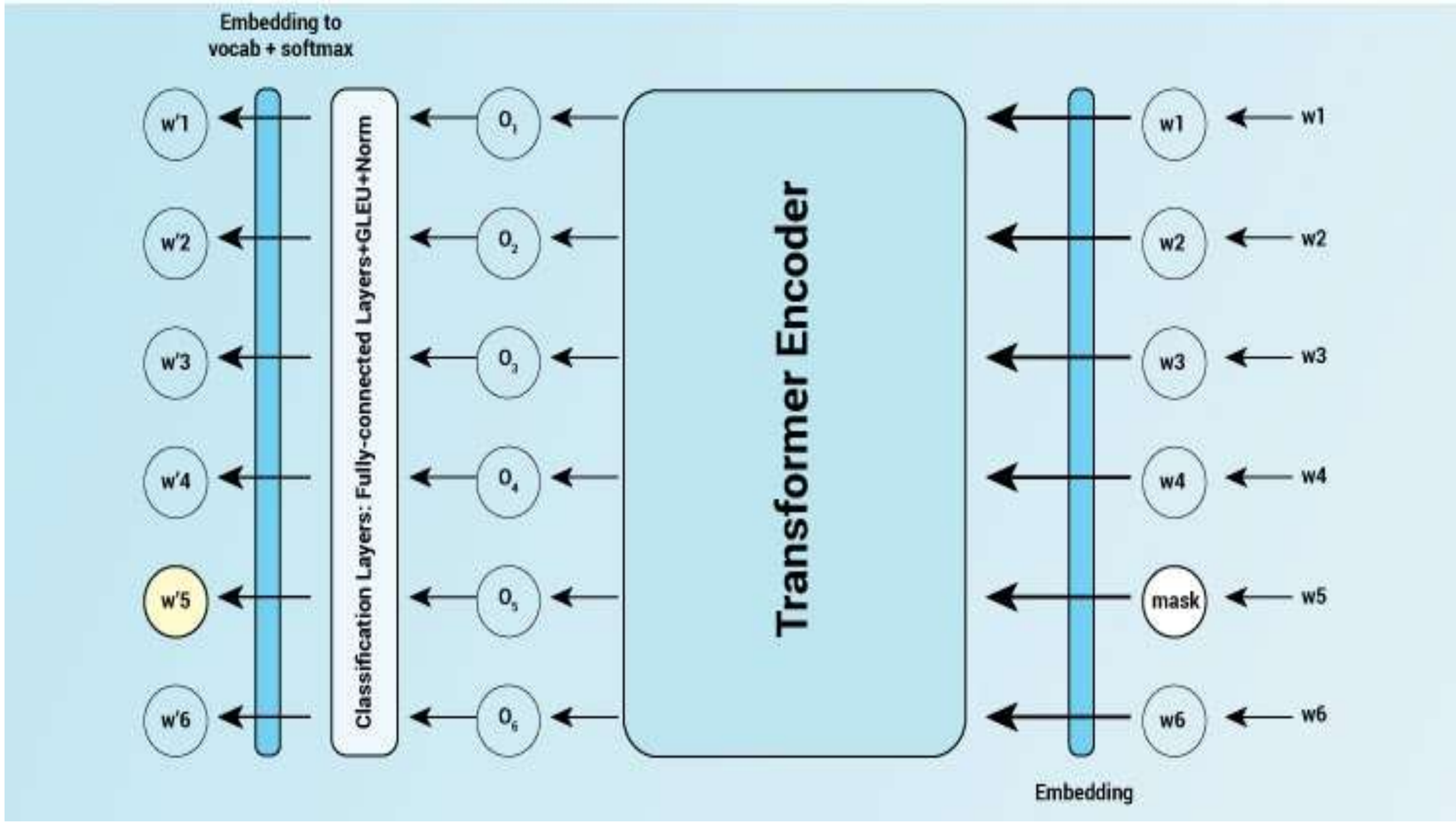
  `Where do water droplets collide with ice crystals to form precipitation?`

- Input Paragraph:

  `... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals within a cloud. ...`

- Output Answer:

  `within a cloud`



Mathematical process

The Neural Attention Mechanism: Neural attention is a neural network component prevalent in contemporary state-of-the-art NLP models such as BERT.

Given a query q and a set of n items $\{X_1, X_2, \ldots \ldots \ldots X_n\}$, the attention mechanism induces a probability distribution α over the item set and then produces an expectation (weighted sum) of the items as the output. Intuitively, attention makes a soft selection of which item X_1 is the best fit for the query q. More formally, each item X_i represented by two vectors: key k_i and value v_i.

The key k_i is used to determine the distribution $\boldsymbol{\alpha}$ via an inner product with the query vector q, normalized by a softmax:

$$\boxed{\alpha_i = \frac{\exp q^\top k_i}{\sum_{\ell=1}^{n} \exp q^\top k_\ell}.}$$

Mathematical process

The output y is then the expectation over the value vectors v_1

$$y = \sum_{i=1}^{n} \alpha_i v_i.$$

Method: Attention Heads as Simple Classifiers.

if α(w,h) denotes the attention distribution of head h when BERT is run over the sequence of words $w=\{w_1, w_2, \dots \dots \dots . w_n\}$ we find the most-attended-to word $w_{\text{argmax}_i \alpha(w,h)_i^j}$

for each position $1 \le j \le n$. If $S_l(w) = \{j : \sum_{i=1}^{n} l(w_i, w_j) > 0\}$ is the subset of the input expressing the annotated relationship, the precision score for the head is computed as

$$\text{precision}(h) = \frac{1}{N} \sum_{w \in \text{corpus}} \sum_{j \in S_l(w)} l\left(w_{\text{argmax}_i \alpha(w,h)_i^j}, w_j\right),$$

Difference with Other Neural Networks

- ❑ BERT uses transformers archtecture of neural network so parallelization can be very helpful whereas the other (ELMO and ULMfit) uses LSTM

- ❑ BERT is purely Bi-directional, GPT is unidirectional and ELMo is semi-bidirectional.

- ❑ GPT is trained on the BooksCorpus (800M words); BERT is trained on the BooksCorpus (800M words) and Wikipedia (2,500M words).

- ❑ For sentiment analysis or question answering tasks, to use BERT, the users have to train the model on a separate layer on sentence encodings. However, GPT-3 uses a few-shot learning process on the input token to predict the output result.

Difference with Other Neural Networks

- ❑ While the transformer includes two separate mechanisms — encoder and decoder, the BERT model only works on encoding mechanisms to generate a language model; however, the GPT-3 combines encoding as well as decoding process to get a transformer decoder for producing text.

- ❑ Being trained on 175 billion parameters, GPT-3 becomes 470 times bigger in size than BERT-Large.

- ❑ BERT trains the language model transformers in both directions while the OPEN AI GPT context trains it only left to right.

- ❑ While BERT requires an elaborated fine-tuning process where users have to gather data of examples to train the model for specific downstream tasks, GPT-3's text-in and text-out API allows the users to reprogram it using instructions and access it.

Difference with Other Neural Networks

- While BERTs use transformers to train language models, LSTMs are used to train Language models and finetune for classification

- BERT, in contrast to Bidirectional LSTMs, can observe the entire sequence at once, in effect allowing to condition the hidden state on the entire sequence is probably contributing to this.

- While GPT-3 is commercially available via an API, but not open-sourced, BERT has been an open-source model since its inception that allows users to fine-tune it according to their needs.

Text Generation with Bert

BERT based Text Generation applies one of the two pretraining steps of BERT, masked word prediction, for text generation. Masked word prediction in BERT pretraining looks like:

```
Masked input: the man went to the [MASK] .
Prediction: [MASK] = store
```

Actually, the model outputs confidence for every word in the dictionary. Thus, by choosing the “incorrect” word with the highest confidence, one can replace a word in a sentence hopefully without any grammatical or semantic incorrectness.

Here is a single step of word replacement at a random position.

```
Input: the man went to the store .
Masked input: the man went to the [MASK]
Output: the man went to the office .
```

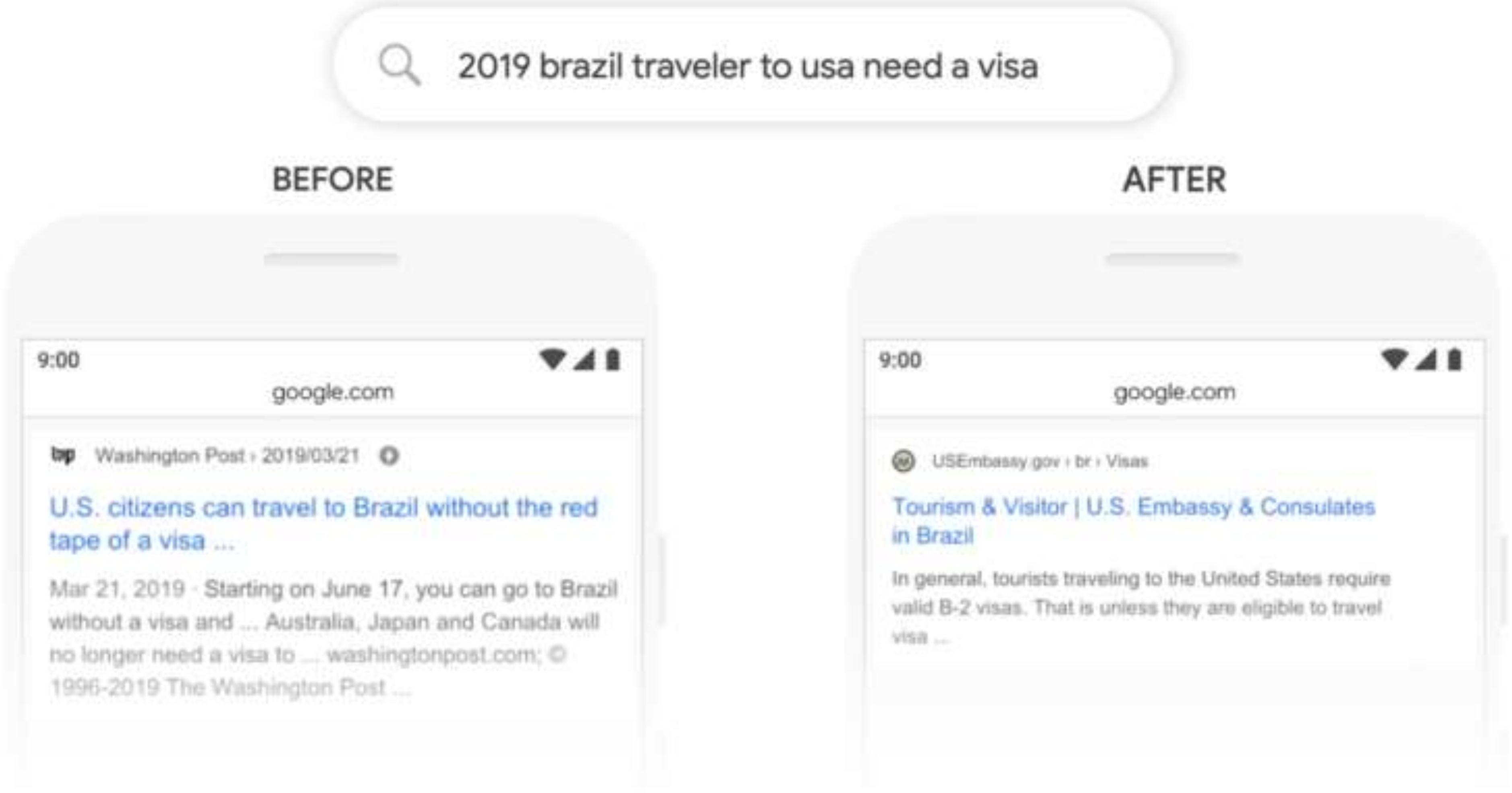
Bert in Search Engine

- Google Search has begun to use Google's transformer neural network **BERT** from 2019 for search queries in over 70 languages.

- The introduction of transformer neural networks to Google Search means that queries where words such as **'from'** or **'to'** affect the meaning are better understood by Google. Users can search in more natural English rather than adapting their search query to what they think Google will understand.

An example from Google's blog is the query "2019 brazil traveler to USA need a visa." The position of the word 'to' is very important for the correct interpretation of the query. The previous implementation of Google Search was not able to pick up this nuance and returned results about USA citizens traveling to Brazil, whereas the transformer model returns much more relevant pages.

Bert in Search Engine



2019 brazil traveler to usa need a visa

BEFORE

9:00

google.com

Washington Post › 2019/03/21

U.S. citizens can travel to Brazil without the red tape of a visa ...

Mar 21, 2019 · Starting on June 17, you can go to Brazil without a visa and ... Australia, Japan and Canada will no longer need a visa to ... washingtonpost.com; © 1996-2019 The Washington Post ...

AFTER

9:00

google.com

USEmbassy.gov › br › Visas

Tourism & Visitor | U.S. Embassy & Consulates in Brazil

In general, tourists traveling to the United States require valid B-2 visas. That is unless they are eligible to travel visa ...

Thank You