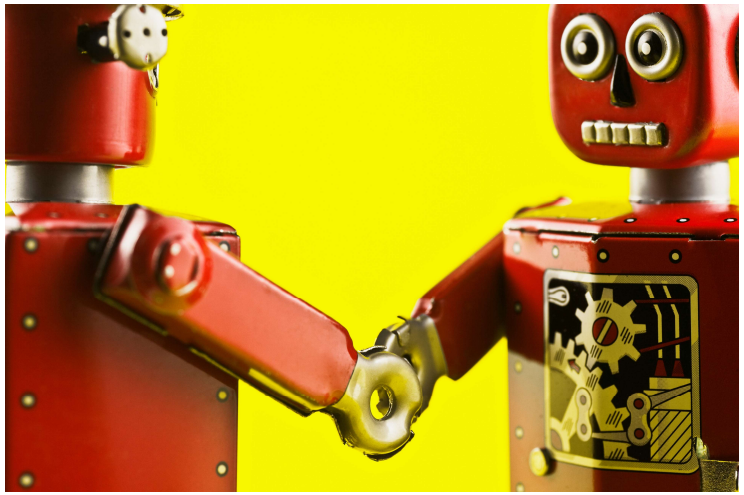
Maestría en Inteligencia Artificial Aplicada (MNA)

Modelo BERT

Procesamiento de Lenguaje Natural (NLP)

Luis Eduardo Falcón Morales

Modelo Transformer pre-entrenado BERT



Otra de las fortalezas y gran impacto que han generado los modelos Transformer es el escalamiento a nuevas tareas diferentes a las que fue entrenado.

En particular, la traducción de textos es una de las tareas con las cuales tienen que dedicar tiempo las organizaciones hoy día, por la gran diversidad de información que circula de manera global.

BERT : Bidirectional Encoder Representations from Transformers

Training:

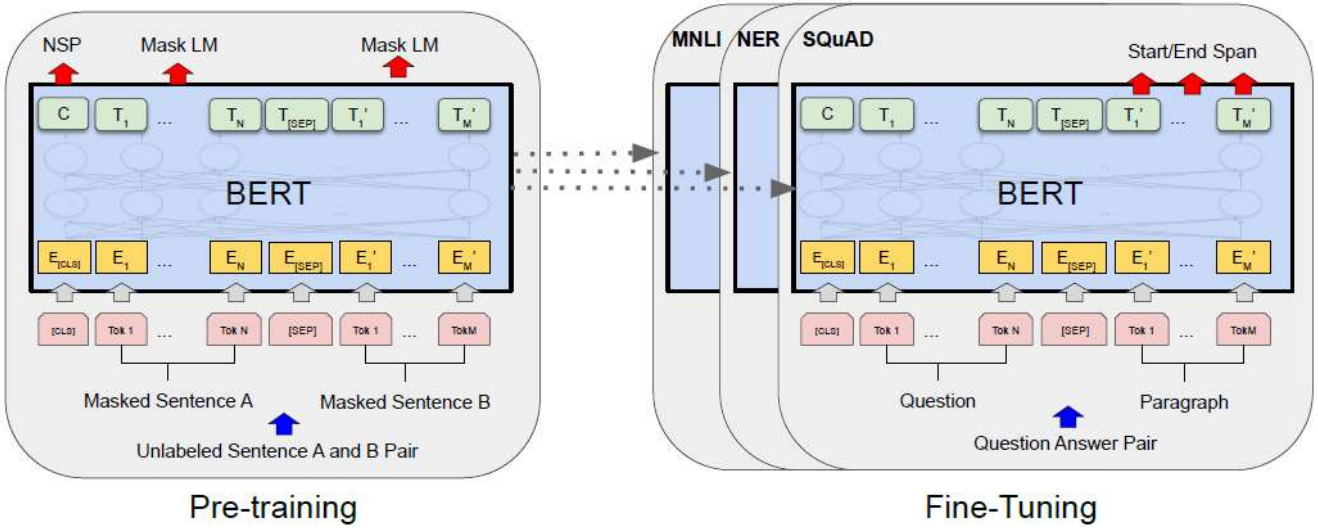
Wikipedia + TorontoBookCorpus

24 May 2019

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Jacob Devlin **Ming-Wei Chang** **Kenton Lee** **Kristina Toutanova**
Google AI Language
{jacobdevlin,mingweichang,kentonl,kristout}@google.com

https://arxiv.org/abs/1810.04805



BERT es un tipo de transformer, pero solo usando la parte del Encoder.

BERT

Pre-trained BERT model

https://github.com/google-research/bert

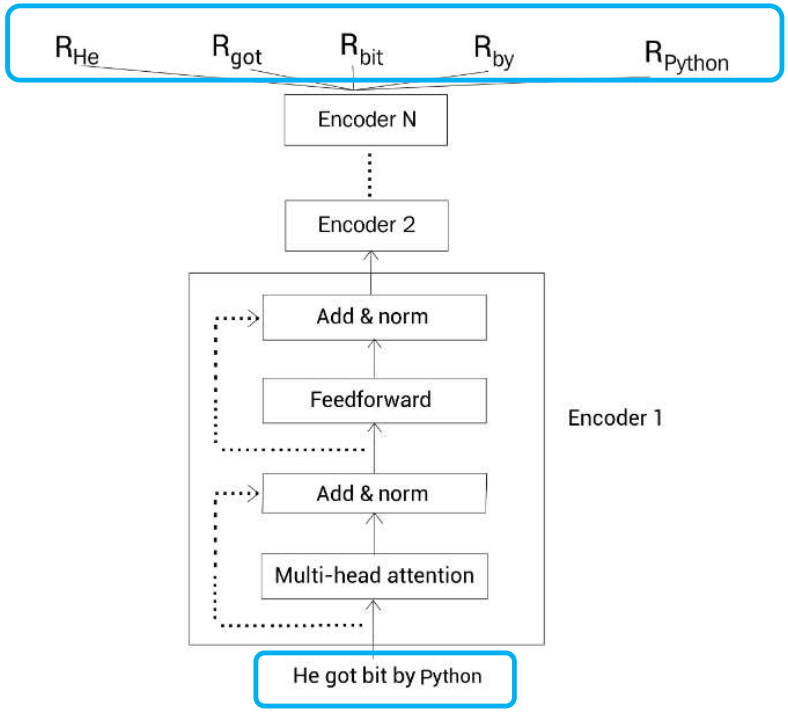
Algunas tareas que se pueden realizar con el BERT pre-entrenado:

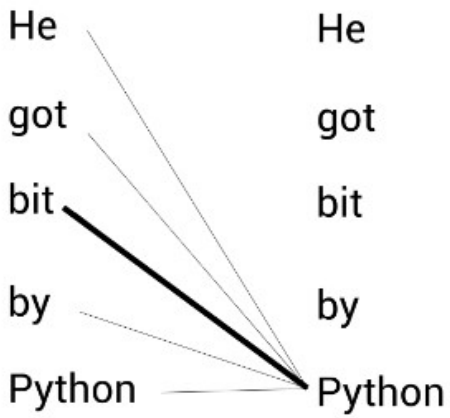
- Feature extractor
- Extract embeddings
- Sentiment-Analysis
- Question & Answering (QA)
- Summarization
- Masked Language Modeling
- Named Entity Recognition (NER)
- Text Classification

Una de las ventajas de BERT es que es un modelo pre-entrenado de vectores embebido basados en contexto, a diferencia de por ejemplo 2ord2vec o fastText que son libres de contexto.

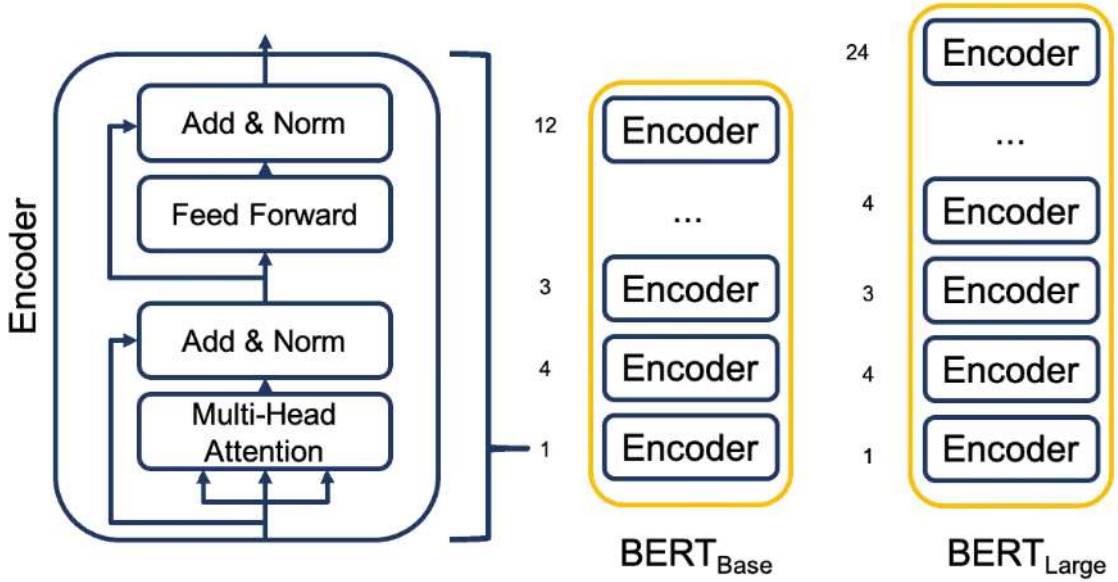
Es decir, BERT genera vectores embebidos (representation) dinámicos basados en el contexto, a partir de los enunciados de entrada. De ahí el nombre: Encoder Representation from Transformer.

Además es bidireccional, ya que tiene acceso a las palabras antes y después de una dada:





Principales configuraciones del modelo BERT



L (Layers) : cantidad de veces que se repite el bloque Encoder.

A (Attention) : cantidad de veces que se repite el módulo (Head) de Attention.

H (Hidden) : Tamaño de la capa oculta.

L=12
A=12
H=768
110 millones
de parámetros

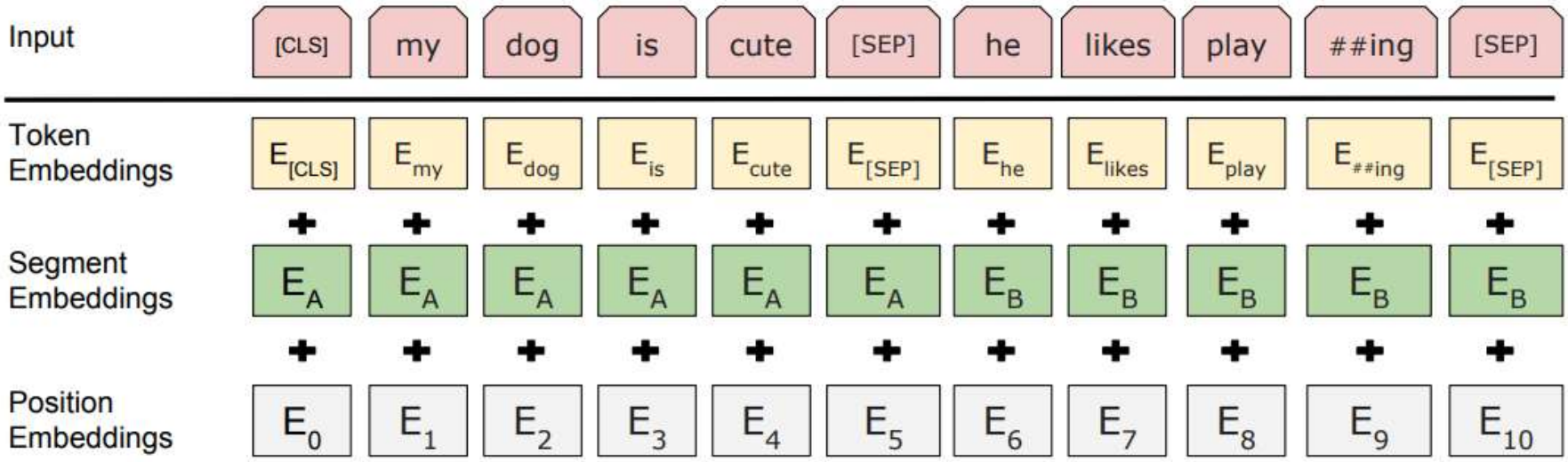
L=24
A=16
H=1024
340 millones
de parámetros

¿Qué significa modelo pre-entrenado?

M1 :
Entrenamos el modelo para una tarea específica con una gran cantidad de datos.

M2 :
Para una nueva tarea, inicializamos el modelo M2 con los pesos del Modelo1 (pre-entrenado). Es decir, en lugar de iniciar entrenado de cero, pre-entrenamos M1 y ajustamos sus pesos (fine-tune) de acuerdo a la nueva tarea.

Existen tres tipos de vectores Embebidos (Embeding) en BERT



https://arxiv.org/abs/1810.04805

BERT : WordPiece tokenizer

- El vocabulario de BERT es de cerca de 30,000 tokens.
- Dado un texto de entrada, si un token aparece en el diccionario, se utiliza como tal.
- Si el token no aparece en el diccionario, se separa en sub-tokens hasta que cada subparte aparezca en el vocabulario.
- En el peor de los casos, cada token se divide en caracteres individuales.
- Este concepto permite manejar cualquier token/palabra de entrada (out-of-vocabulary: OOV)

[let, us, start, pre, ##train, ##ing, the, model]

Modelado de Lenguaje : Language Modeling

Sin música, la vida sería un error.
F. Nietzsche

Modelado de lenguaje: dada una secuencia de tokens, se desea predecir el siguiente token.

Sin música, la [MASK] sería un error.

Existen dos tipos de modelado de lenguaje:

Auto-regresivo:

- Predicción hacia adelante (de izquierda a derecha)

Sin música, la [MASK]

- Predicción hacia atrás (de derecha a izquierda)

[MASK] sería un error.

Auto-encoding: BERT es autoencoding

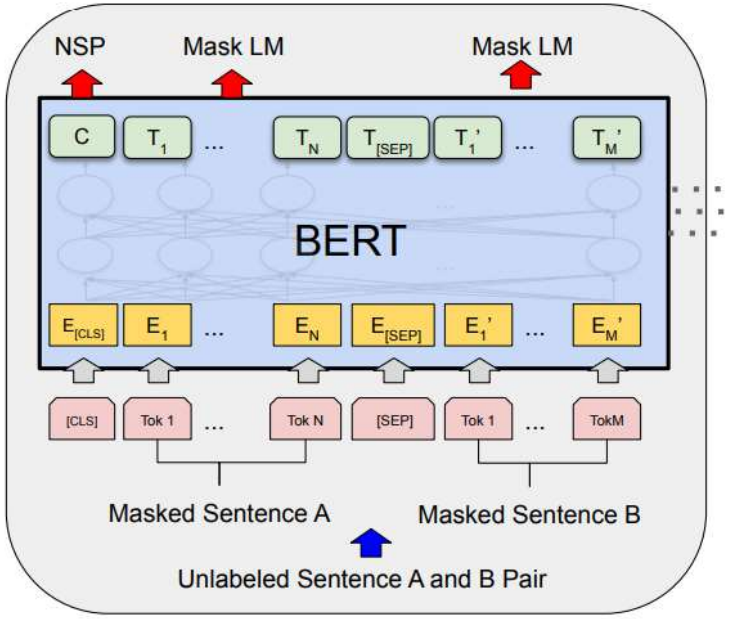
- Predicción leyendo de hacia adelante y hacia atrás (es decir, de izquierda a derecha y de derecha a izquierda)

Sin música, la [MASK] sería un error.

Técnicas de pre-entrenamiento

BERT es pre-entrenado con base a dos tareas:

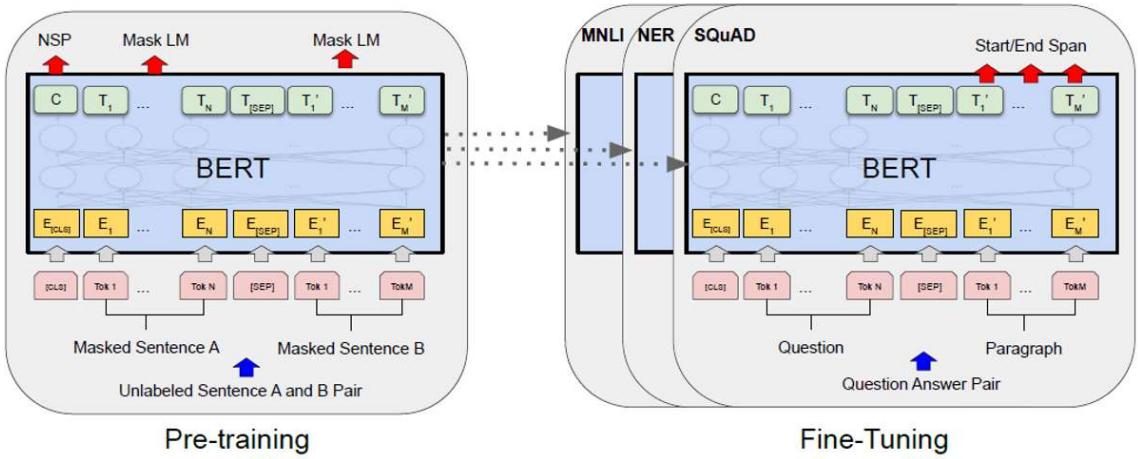
- Modelado de Lenguaje Enmascarado (Masked Language Modeling – MaskLM)
- Predicción del Siguiente Enunciado (Next Sentence Prediction – NSP)



Pre-training

BERT se entrena con ambas tareas al mismo tiempo.

A partir de los modelos pre-entrenados, se pueden generar nuevas tareas mediante ajuste de parámetros (fine-tuning):



Solamente con las tareas de la etapa de pre-entrenamiento necesito los grandes corpus para generar los pesos del modelo y de ahí con ajuste de parámetros obtenemos los pesos para nuevas tareas.

Modelado de Lenguaje Enmascarado (Masked Language Modeling – MaskLM)

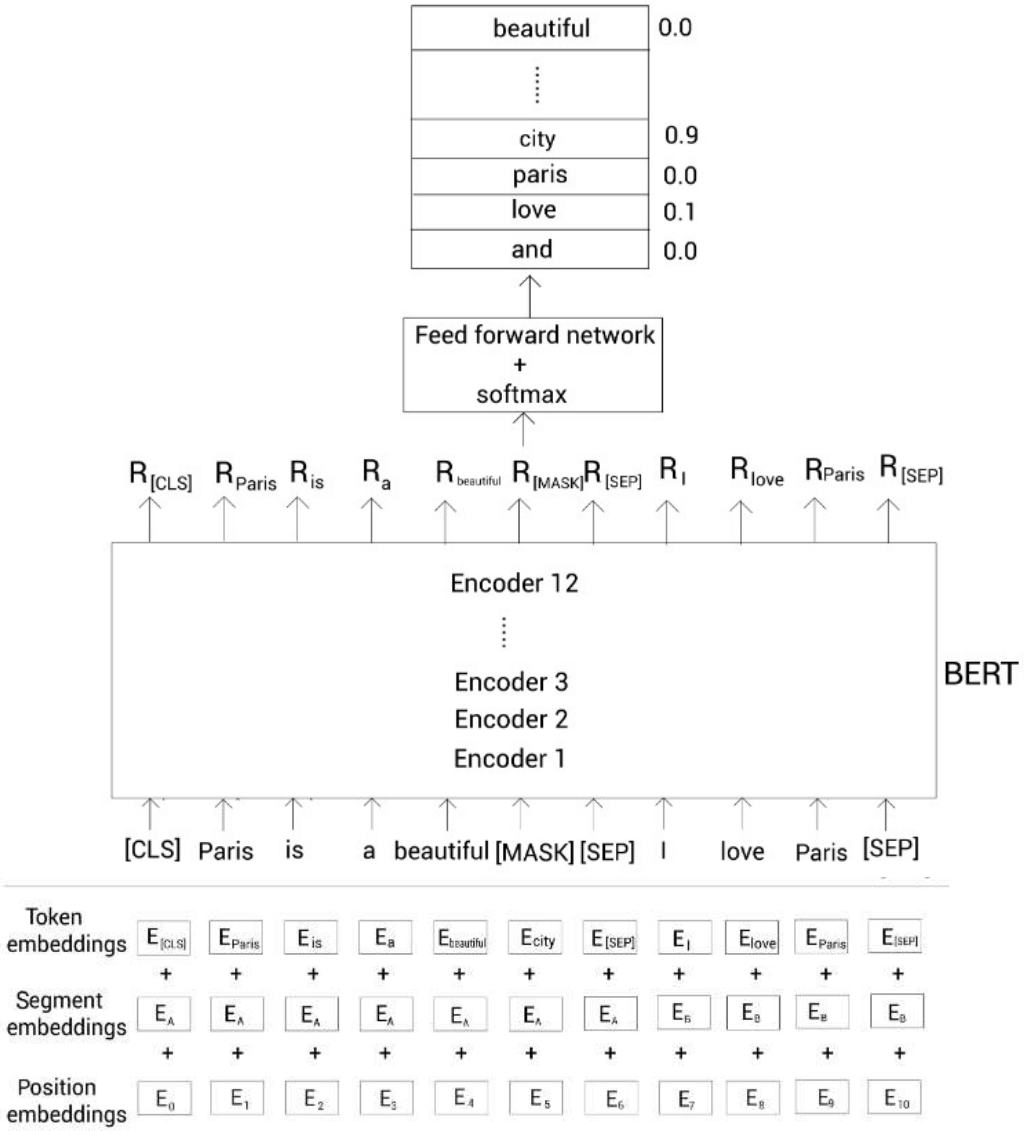
En el modelado autoencoding, BERT enmascara el 15% de los tokens de un texto de entrada y tratará de predecirlos a la salida de los Encoders:

De este 15% de tokens enmascarados:

- 80% de las veces se enmascara con el token [MASK].
- 10% de las veces se sustituye por cualquier otro token del vocabulario.
- 10% de las veces no se hace sustitución alguna, se queda el enunciado igual.

NOTA – Whole Word Masking:

Si un sub-token es enmascarado, todos los subtokens relacionados también se enmascaran y todos se contabilizan para el 15% de tokens enmascarados: ##ferro, ##carril → [MASK], [MASK].


beautiful
0.0
city
0.9
paris
0.0
love
0.1
and
0.0
Feed forward network
+
softmax
R[CLS] RParis Ris Ra Rbeautiful R[MASK] R[SEP] RI Rlove RParis R[SEP]
Encoder 12
Encoder 3
Encoder 2
Encoder 1
BERT
[CLS] Paris is a beautiful [MASK] [SEP] I love Paris [SEP]
Token embeddings
Segment embeddings
Position embeddings

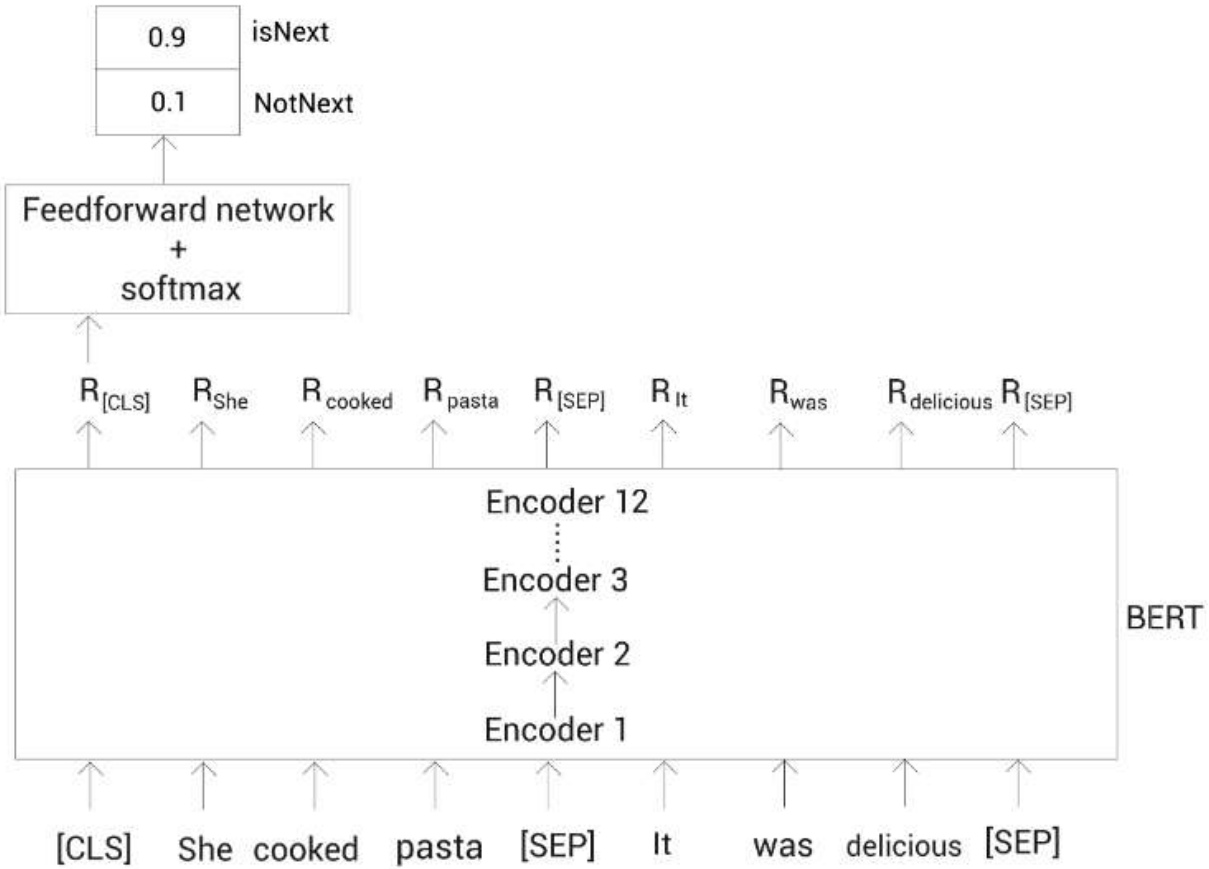
Predicción del Siguiente Enunciado (Next Sentence Prediction – NSP)

NSP se considera una tarea de clasificación binaria.

Como entrada al modelo BERT se tienen dos enunciados y se trata de predecir si el segundo se puede considerar como el enunciado que sigue al primero, dentro de un mismo contexto.

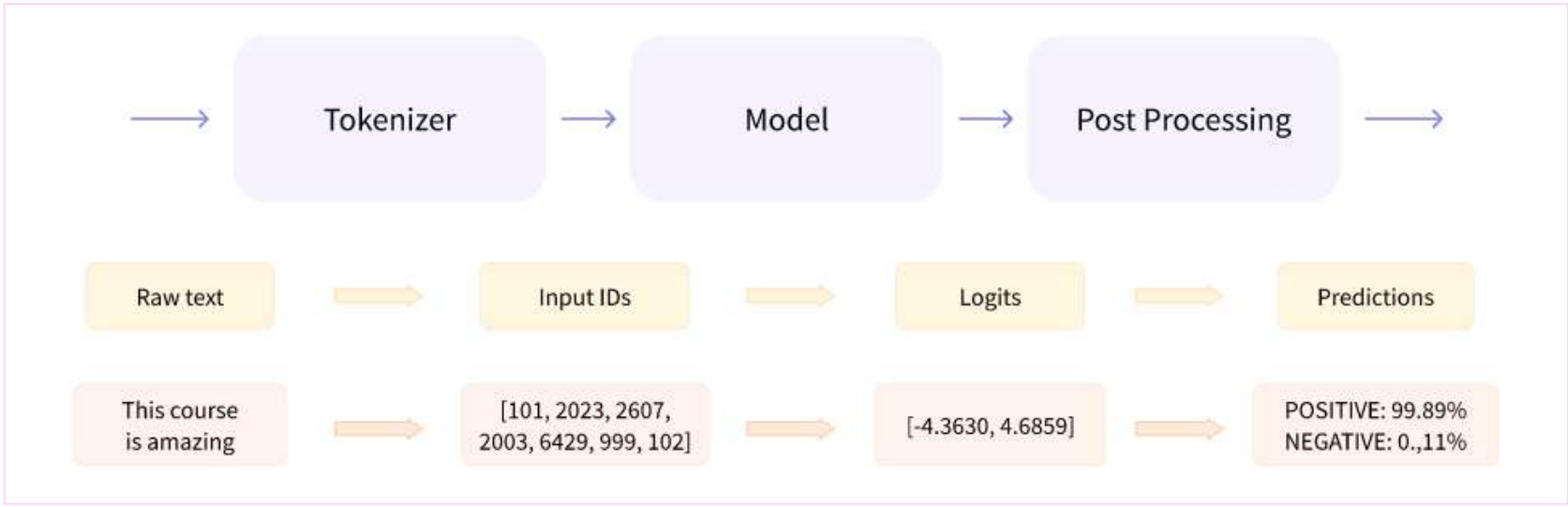
• El niño está en el jardín. • Está jugando con el perro.	• El niño está en el jardín. • El camión es de pasajeros.
isNext	**notNext**

Esta tarea será de interés para modelos de Question&Answering y de generación de texto.


0.9
isNext
0.1
NotNext
Feedforward network
+
softmax
R[CLS]
RShe
Rcooked
Rpasta
R[SEP]
RIt
Rwas
Rdelicious
R[SEP]
Encoder 12
Encoder 3
Encoder 2
Encoder 1
BERT
[CLS]
She
cooked
pasta
[SEP]
It
was
delicious
[SEP]

Hugging Face

https://huggingface.co/

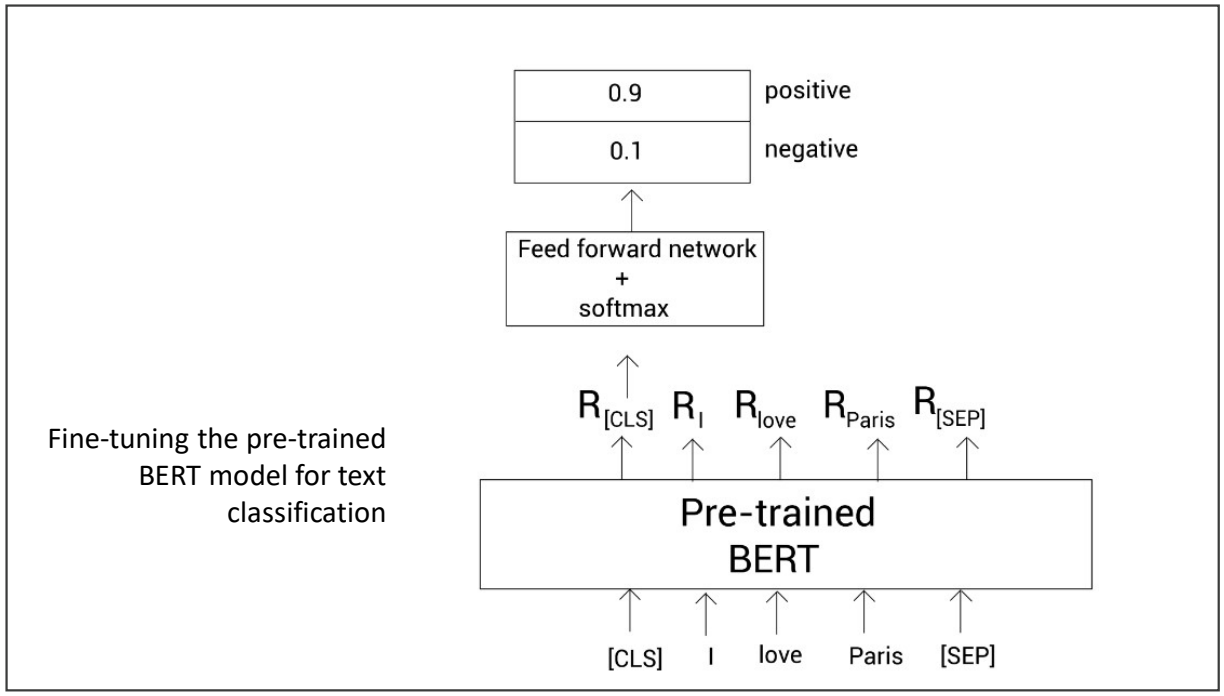


HF es una organización que se propone democratizar la IA a través del procesamiento del lenguaje natural y sobre todo con los modelos Transformers, al crear una plataforma de código abierto a través de la cual se accede a modelos pre-entrenados de última generación.

BERT model as a feature extractor vs fine-tuning

Fine-tune the pre-trained BERT model for text classification (sentiment análisis) task:

Text classifier: Podemos extraer el R_[CLS] del modelo BERT para cada enunciado (comentario/twitter) y estas serán las entradas de una red neuroanl a entrenar, digamos.



Fine-tuning the pre-trained BERT model for text classification

Es importante notar que en este proceso, los pesos de BERT pre-entrenado no se ajustan/modifican. Solamente los vectores emebidos de los enunciados se ajustan.

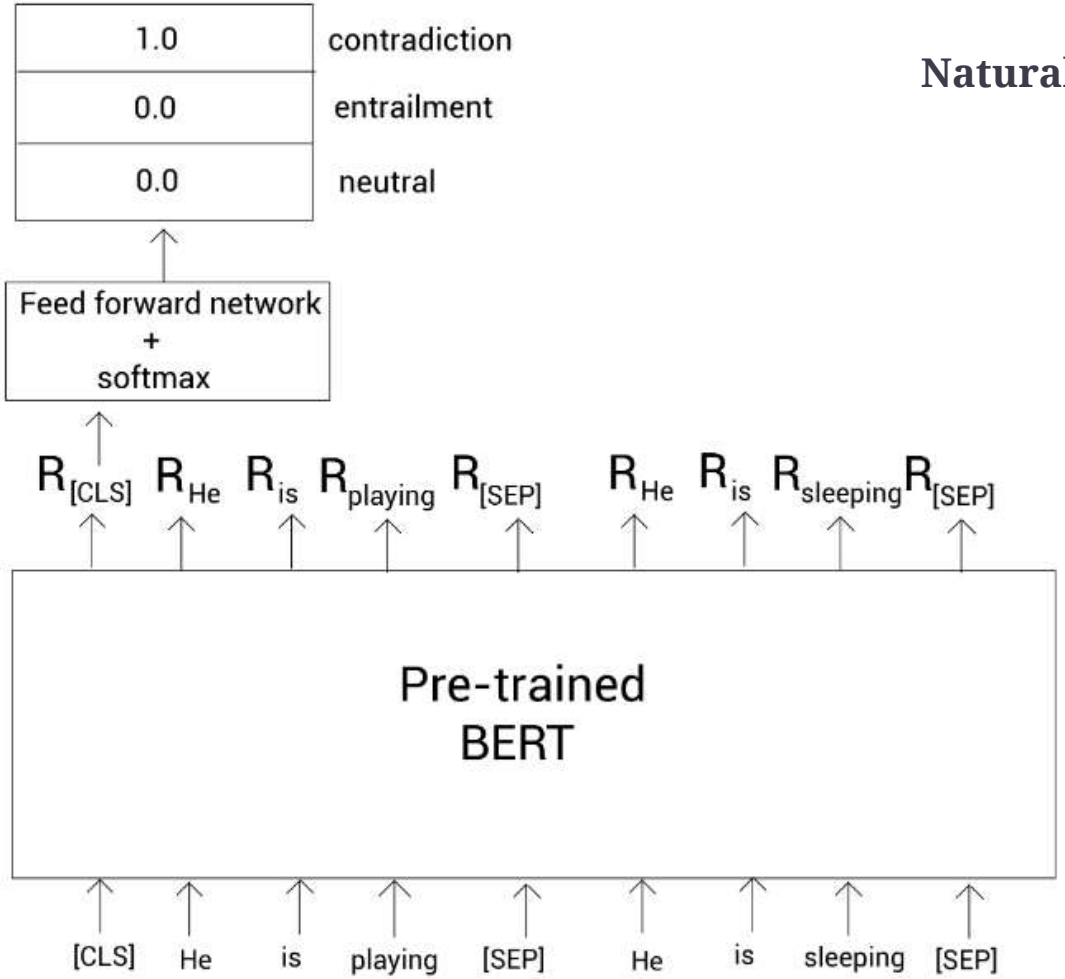
Summarization / Resumen

La manera en que funciona este es como sigue:

Se tiene un texto inicial que se quiere resumir:
Se extraen párrafos que pudieran ser el resumen y se busca qué tanto ambos son equivalentes, para que pueda ser parte del resumen.

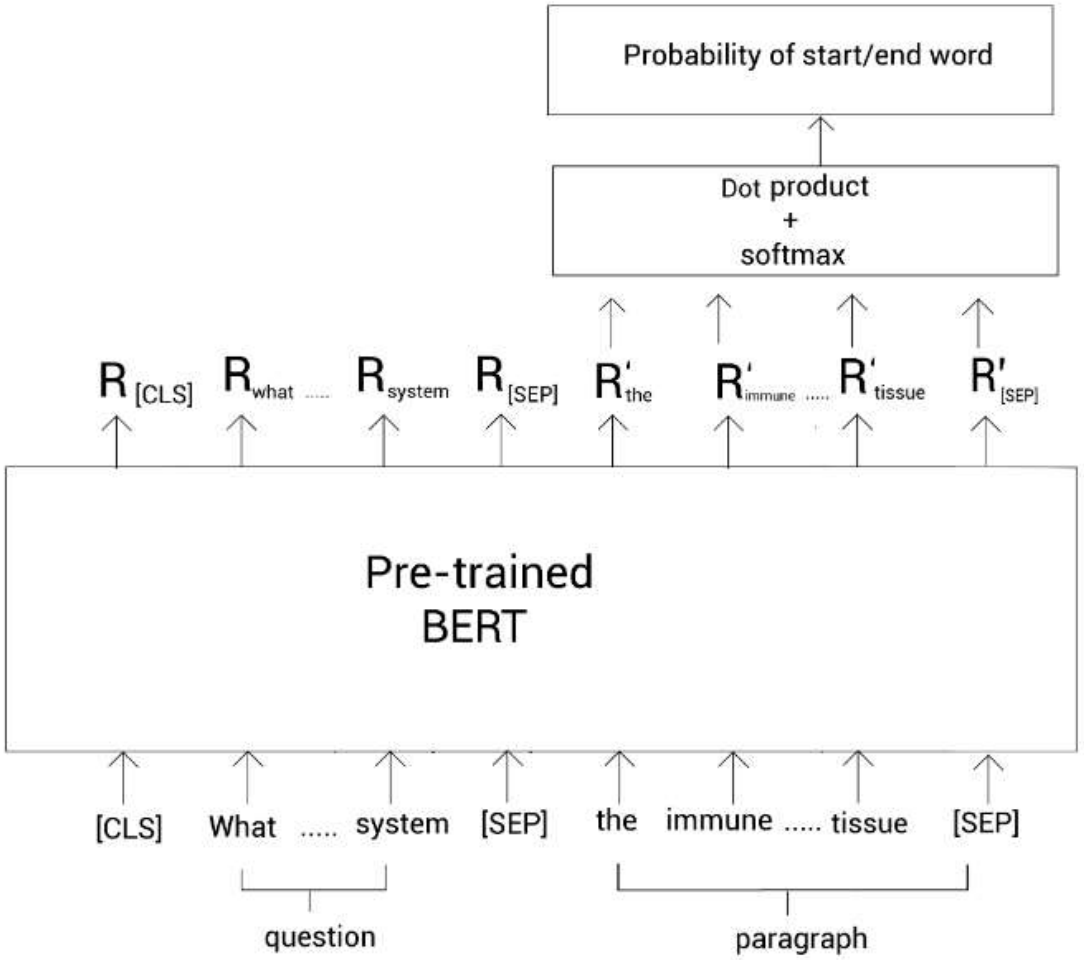
Es decir, esto es equivalente al NextSentencePrediction (NSP) que usa BERT.

Natural Language Inference (NLI)

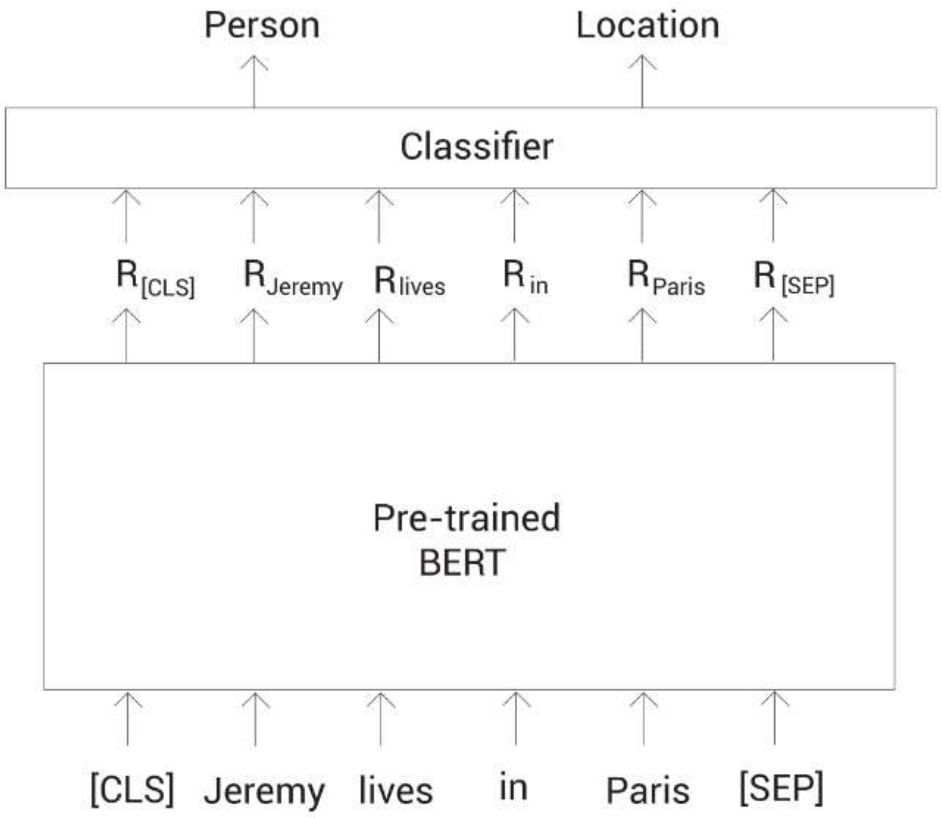


Fine-tuning the pre-trained BERT model for NLI

Como ajustar parámetros para el BERT :Question&Answering



Named Entity Recognition (NER)



Fine-tuning the pre-trained BERT model for NER