



**Data Science
Academy**

www.datascienceacademy.com.br

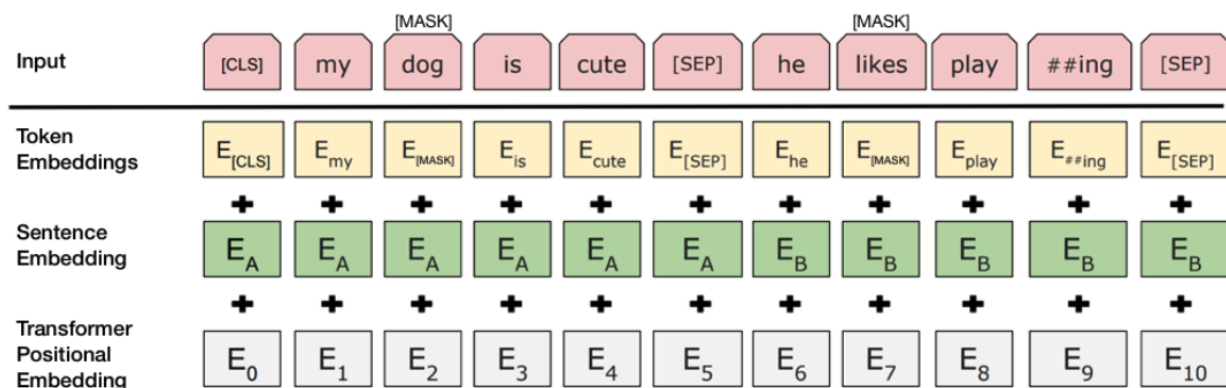
Processamento de Linguagem Natural

Next Sentence Prediction (NSP)

No processo de treinamento do BERT, o modelo recebe pares de sentenças como entrada e aprende a prever se a segunda sentença do par é a sentença subsequente no documento original. Durante o treinamento, 50% das entradas são um par no qual a segunda sentença é a sentença subsequente no documento original, enquanto nos outros 50% uma sentença aleatória do corpus é escolhida como a segunda sentença. A suposição é que a sentença aleatória será desconectada da primeira sentença.

Para ajudar o modelo a distinguir entre as duas frases no treinamento, a entrada é processada da seguinte maneira antes de entrar no modelo:

- Um token [CLS] é inserido no início da primeira frase e um token [SEP] é inserido no final de cada frase.
- Uma embedding de sentença indicando a Sentença A ou a Sentença B é adicionada a cada token. Os embeddings de sentença têm conceito semelhante aos embeddings de token com um vocabulário de 2.
- Uma embedding posicional é adicionada a cada token para indicar sua posição na sequência. O conceito e a implementação de embedding posicional são apresentados no artigo Transformer (link nas referências).



Para prever se a segunda frase está realmente conectada à primeira, são executadas as seguintes etapas:

- Toda a sequência de entrada passa pelo modelo Transformer.
- A saída do token [CLS] é transformada em um vetor em formato 2×1 , usando uma camada de classificação simples (matrizes aprendidas de pesos e bias).
- Calcula-se a probabilidade de IsNextSequence com softmax.

Ao treinar o modelo BERT, o Masked LM e a previsão de próxima sentença são treinados juntos, com o objetivo de minimizar a função de perda combinada das duas estratégias.



Referências:

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
<https://arxiv.org/abs/1810.04805>

Transformer: A Novel Neural Network Architecture for Language Understanding
<https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>