BERT notebook actividades
Fernanda Pérez A01742102

Del notebook **notebooke72f36b4c0 (Text Classification Using Transformer Networks (BERT)**

Incluya una breve descripción de la estructura del pipeline que ejecuta este codigo:

Este código hace una implementación de un pipeline para poder hacer una clasificación de textos como la vez pasada, solo que en esta ocasión está usando el modelo BERT (Bidirectional Encoder Representations from Transformers), "un algoritmo de Google basado en Inteligencia Artificial (Machine Learning) y que está focaliza en el procesamiento natural del lenguaje. Utiliza PLN (Procesamiento del lenguaje natural) que básicamente es un sistema computacional de comprensión de lenguaje" ("Luis Salazar Jurado," 2020). En el desarrollo de la actividad los textos se tokenizan y se tienen que preparan para poder ser procesados por un modelo de aprendizaje profundo preentrenado. Este modelo seleccionado, BERT , sirve para tareas de clasificación de secuencias, en donde cada entrada va a recibir una etiqueta que corresponde a una categoría. El código de la actividad incluye: entrenamiento, evaluación y predicción, y se emplean las métricas de precisión y exactitud para poder medir y calificar su desempeño.

El pipeline que se ejecutó en el código para la realización de esta actividad fue que este comienza primeramente con el preprocesamiento de datos, en donde los textos se tokenizan utilizando un tokenizador compatible con BERT para lograr generar : vectores de entrada o sea input_ids, máscaras de atención , o sea attention_mask y los tipos de tokens , o sea token_type_ids.Despues , se inicializa un modelo BERT previamente entrenado o sea que ya esta preentrenado, modificado con una capa final para clasificación que ajusta las representaciones de texto a las etiquetas de salida. El modelo se logra entrenar haciendo uso de un objeto 'Trainer' que de manera automáticamente gestiona : las iteraciones de entrenamiento, la evaluación en cada época y tambien el cálculo de métricas : precisión y exactitud.Y ya por ultimo evaluamos el modelo en un conjunto de prueba para generar predicciones y métricas de rendimiento : precision, recall, f1-score, esto con el fin de obtener una mejor idea de los resultados y evaluar el desempeño que se logró con el sistema de clasificación de textos.

2.
-
Realice el entrenamiento de clasificación de reseñas de películas IM
BD basada en una implementación de TensorFlow en el siguiente notebook.

Deberá realizar a 4 experimentos

Deberá realizar a 4 experimentos:

(a) Nsamp = 1000, maxtokens = 50, maxtokenlen = 20
(b) Nsamp = 1000, maxtokens = 100, maxtokenlen = 100
(c) Nsamp = 1000, maxtokens = 200, maxtokenlen = 200
(d) Nsamp = 1000, maxtokens = 230, maxtokenlen = 200

Experimento	Train Accuracy	Val Accuracy	Train Loss	Val Loss
a)	0.7589	0.695	0.4823	0.5891
b)	0.8127	0.8023	0.4234	0.4517
c)	0.8345	0.7514	0.3912	0.5048
d)	0.8456	0.7932	0.3765	0.4376

Los resultados que se obtuvieron con los 4 experimentos nos muestran que, conforme aumenta el número de tokens, la precisión en entrenamiento y en validación mejoran, o se que el modelo aprovecha mejor los embeddings contextuales de BERT. Al analizar los resultados vemos que el experimento es el que tiene el mejor rendimiento con un Validation Accuracy de 0.7932, demostrando que configuraciones más grandes permiten al modelo capturar mejor las relaciones en los datos, sin caer en un sobreajuste.

Referencias

Luis Salazar Jurado. (2020). Retrieved November 25, 2024, from Consultor SEO Freelance website: https://www.seotecnico.com/bert.html