Tranformers and Large Language Models - Winter School - PUCP

Parte II: Modelos Preentrenados

Sebastián Vallejo Vera [1]

August 12, 2025

[1] Dept. of Political Science – Western University

Índice

Transformers en Encoders

Modelos Preentrenados i

- La arquitectura Transformers *incrusta?* (embed) tokens con significado, creando representaciones precisas (es decir, embeddings).
- Al igual que con los embeddings, las representaciones son relacionales, proporcionan significados a los tokens en relación con otros tokens.
- Esto significa que, cuantos más tokens podamos procesar a través de la arquitectura, más precisas serán estas representaciones.

Modelos Preentrenados ii

- Modelos preentrenados como BERT, aprovechan la arquitectura Transformers, pero también necesitan (1) información y una forma de (2) aprender de esa información.
- Ellos aprenden, como aprendimos previamente, mediante *masking*.
- Obtener información es relativamente fácil (y poco ético) y computacionalmente costoso:
  - BERT usa 11,038 libros del Toronto BookCorpus y toda la Wikipedia en inglés–un total de 16GB de texto (Devlin et al., 2018).
  - RoBERTa usa los mismos datos que BERT y añade más datos de Common Crawl (CC-News/Stories) y Open WebText para un total de 160GB de texto (Liu et al., 2019).
  - RoBERTa multilingüe, XLM-R, fue entrenado usando Wikipedia para todos los idiomas y datos de Common Crawl (Conneau et al., 2019).

Bidirectional Encoder Representations from Transformers

- BERT de Google AI, RoBERTa y XLM-RoBERTa de Facebook AI, y DeBERTa de Microsoft son los encoders de un modelo Transformers.
- BERT significa 'Bidirectional Encoder Representations from Transformers'. RoBERTa significa 'Robustly optimized BERT approach' y XLM-RoBERTa significa "Cross-lingual RoBERTa."
- Como nota, *bidireccional* apunta a su habilidad para leer texto hacia adelante *y* hacia atrás, relacionando cada palabra con todas las palabras en una oración.
- Y debido al mecanismo de atención, las representaciones cambian según el contexto.

Recuerda el Mecanismo del Encoder

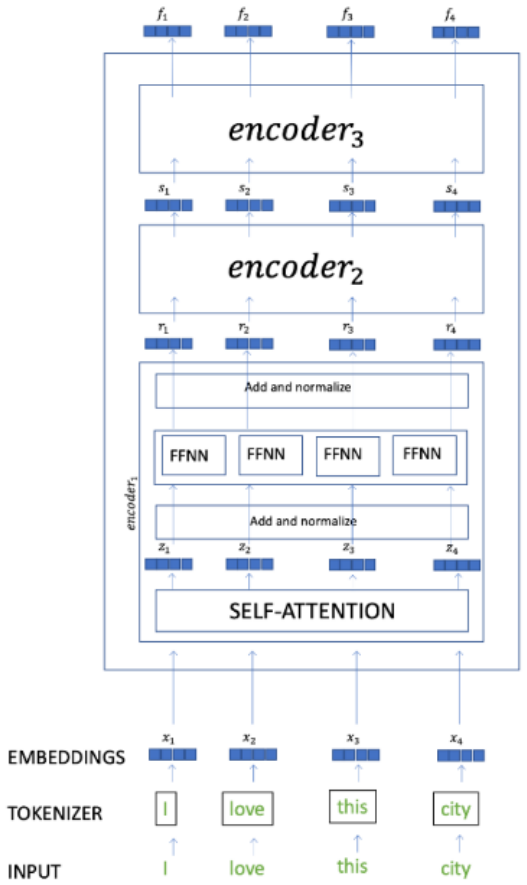

Figure 1: Diagram of a three-stack encoder of a Transformers model. Input text is tokenized and given an initial embedding (vectorized representation) simplified in our figure as x_1 through x_4. The initial embeddings are transformed as they enter the first $encoder_1$. In it, the self-attention mechanism updates the embeddings (z_1 through z_4), which are then passed through a feed-forward neural network. They exit the encoder as a more accurate set of embeddings (r_1 through r_4). The process is repeated for all encoders in the neural network. For example, pre-trained BERT-base models use 12 encoder layers.

¿Cómo se Entrenan los Modelos Preentrenados?

- BERT-base consta de 12 capas de encoder y 12 cabezas de atención, y 24 capas de encoder y 16 cabezas de atención en su configuración BERT-large (Ravichandiran, 2021).
    - Las cabezas de atención repiten sus cálculos múltiples veces en paralelo. La cabeza de atención divide sus parámetros Query, Key y Value en N partes y pasa cada división independientemente a través de una cabeza separada. Todos estos cálculos similares de atención se combinan luego para producir un puntaje final de atención.
- BERT-base produce vectores de palabras de longitud 768, mientras que el modelo BERT-large produce vectores de palabras de longitud 1,024 (lo mismo aplica para las versiones base y large de RoBERTa y XLM-R).

Fine-Tuning y Entrenamiento Adicional

Fine-Tuning de un Modelo Preentrenado

- Una vez que un modelo ha sido preentrenado (por ejemplo, BERT), podemos modificar la última capa para realizar tareas de clasificación. Esto se llama fine-tuning.
- Para ello, necesitamos un *training set* y los datos objetivo (es decir, datos a predecir).
- La mayoría de las consideraciones al entrenar modelos supervisados aplican, aunque los Transformers han mostrado varios beneficios respecto a enfoques previos:
  - Mejor comprensión de vocabulario fuera de contexto.
  - Mayor robustez fuera de dominio (*out-of-domain robustness)*(Hendrycks et al., 2020).
  - Se requieren menos observaciones por categoría para un buen desempeño.

¡Al código!

Fine-Tuning: Otras Consideraciones

- Diferentes capas significan diferentes cosas (Jawahar, Sagot and Seddah, 2019; Zhang et al., 2025). No está claro qué significa cada capa, o dónde está la representación más precisa de un token particular, pero vale la pena considerarlo. Sin embargo, en general, el consenso es que cuantas más capas, mejor.
- Al predecir etiquetas, usaremos una regresión logística softmax, pero podemos usar una regresión ordinal si ese es el caso.

Entrenamiento Adicional

¿Qué pasa si tienes un corpus con lenguaje especializado que es poco probable que aparezca en el mismo contexto que los datos usados para preentrenar BERT y compañía?

Entrenamiento Adicional

¿Qué pasa si tienes un corpus con lenguaje especializado que es poco probable que aparezca en el mismo contexto que los datos usados para preentrenar BERT y compañía? ¿Qué pasa si tengo tokens que son muy informativos pero que no están en los datos usados para preentrenar BERT y compañía?

Entrenamiento Adicional

¿Qué pasa si tienes un corpus con lenguaje especializado que es poco probable que aparezca en el mismo contexto que los datos usados para preentrenar BERT y compañía? ¿Qué pasa si tengo tokens que son muy informativos pero que no están en los datos usados para preentrenar BERT y compañía? Una cosa interesante de los modelos preentrenados es que podemos entrenarlos adicionalmente para mejorar su desempeño Timoneda and Vallejo Vera (2025).

Hay cuatro pasos para entrenar adicionalmente un modelo preentrenado:

1. Añadimos nuevos tokens al modelo original (opcional).
2. (Si 1, entonces) asignamos la representación media de palabras similares a los tokens recién añadidos.
3. Alimentamos un nuevo corpus grande de texto no estructurado (si 1 y 2, que contenga los nuevos tokens) y lo entrenamos nuevamente para mejorar las representaciones de esos tokens.
4. Guardamos el nuevo modelo y lo aplicamos a nuestra tarea de

¡Al código!

Entrenamiento Adicional: Otras Consideraciones

- Podemos entrenar adicionalmente un modelo usando diferentes estrategias de masking (ver Timoneda and Vera, 2025).
- Necesitas MUCHO texto para ver mejoras en el desempeño.
- Necesitas MUCHOS recursos computacionales para entrenar adicionalmente un modelo.

¿Preguntas? ¿Comentarios? ¿Pausa?

References

Conneau, Alexis, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer and Veselin Stoyanov. 2019. "Unsupervised cross-lingual representation learning at scale." *arXiv preprint* .

Devlin, Jacob, Ming-Wei Chang, Kenton Lee and Kristina Toutanova. 2018. "Bert: Pre-training of bidirectional transformers for language understanding." *arXiv preprint* .

Hendrycks, Dan, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan and Dawn Song. 2020. "Pretrained transformers improve out-of-distribution robustness." *arXiv preprint arXiv:2004.06100* .

Referencias ii

Jawahar, Ganesh, Benoît Sagot and Djamé Seddah. 2019. What Does BERT Learn about the Structure of Language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, ed. Anna Korhonen, David Traum and Lluís Màrquez. Florence, Italy: Association for Computational Linguistics pp. 3651–3657.
URL: *https://aclanthology.org/P19-1356/*

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer and Veselin Stoyanov. 2019. "Roberta: A robustly optimized bert pretraining approach." *arXiv preprint arXiv:1907.11692* .

Ravichandiran, Sudharsan. 2021. *Getting Started with Google BERT*. Packt Publishing.

Referencias iii

Timoneda, Joan C and Sebastián Vallejo Vera. 2025. "Behind the mask: Random and selective masking in transformer models applied to specialized social science texts." *PloS one* 20(2):e0318421.

Timoneda, Joan C. and Sebastián Vallejo Vera. 2025. "BERT, RoBERTa or DeBERTa? Comparing Performance Across Transformer Models in Political Science Text." *The Journal of Politics* 00(00):00.

Zhang, Cheng, Jinxin Lv, Jingxu Cao, Jiachuan Sheng, Dawei Song and Tiancheng Zhang. 2025. "Unravelling the semantic mysteries of transformers layer by layer." *The Computer Journal* p. bxaf034.