Transformers

Joseph Tuffit Hadad Piña

March 22, 2025

1 Resumen

En el presente trabajo mostramos la implementación de un traductor neuronal inglés-español basado en la arquitectura Transformer. Se implementaron dos modelos: un modelo base y una versión mejorada con pre embeddings GloVe. Se describe la arquitectura, el proceso de entrenamiento y los resultados obtenidos, mostrando las dificultades propias del desarrollo de sistemas de traducción.

2 Introducción

La traducción automática neuronal representa uno de los avances más significativos en el procesamiento del lenguaje natural. Este proyecto implementa un traductor inglés-español utilizando la arquitectura Transformer, que ha demostrado ser efectiva para tareas de traducción debido a su capacidad para capturar las secuencias de texto.

3 Marco Teórico

3.1 Arquitectura Transformer

La arquitectura Transformer, introducida por Vaswani et al. (2017), revolucionó el procesamiento del lenguaje natural al eliminar la necesidad de redes recurrentes y convolutivas, basándose completamente en mecanismos de atención.

A diferencia de las arquitecturas RNN o LSTM, los Transformers pueden procesar todas las palabras de una oración simultáneamente, lo que permite un entrenamiento más rápido y paralelo. Su arquitectura consiste principalmente de:

- **Codificador (Encoder)**: Procesa el texto de entrada y genera representaciones contextuales.
- **Decodificador (Decoder)**: Genera la traducción utilizando las representaciones generadas por el codificador.

3.2 Mecanismo de Atención

El componente central de la arquitectura Transformer es el mecanismo de atención, específicamente la atención multi-cabeza (Multi-Head Attention). Este mecanismo permite al modelo enfocarse en diferentes partes de la secuencia de entrada para generar cada palabra de salida.

La atención funciona calculando puntuaciones de relevancia entre cada par de palabras, permitiendo que el modelo determine qué palabras están relacionadas entre sí, incluso si están distantes en la secuencia. En la traducción, esto es crucial para capturar correctamente:

- Relaciones gramaticales entre palabras
- Contexto necesario para traducciones precisas
- Correspondencias entre idiomas con estructuras diferentes

La fórmula básica de la atención es:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

donde Q (consultas), K (claves) y V (valores) son transformaciones lineales de las entradas.

3.3 Embeddings de Palabras

Los modelos de traducción necesitan representar las palabras como vectores densos. En este proyecto:

- Se utilizaron inicialmente embeddings entrenados desde cero
- Se implementó una versión mejorada utilizando embeddings pre-entrenados GloVe, que capturan información semántica de grandes corpus de texto

4 Metodología

4.1 Datos

Se utilizó el conjunto de datos de traducción español-inglés disponible en TensorFlow, que contiene:

- 118,964 pares de oraciones en total
- 83,276 pares para entrenamiento (70%)
- 17,844 pares para validación (15%)
- 17,844 pares para prueba (15%)

4.2 Preprocesamiento

El preprocesamiento incluyó:

- Tokenización y normalización del texto
- Adición de tokens especiales `[start]` y `[end]` al español
- Limitación de secuencias a 20 tokens
- Vectorización con vocabulario de 15,000 palabras

4.3 Arquitectura del Modelo

Se implementaron las siguientes clases personalizadas:

- **PositionalEmbedding**: Combina embeddings de tokens con información posicional
- **TransformerEncoder**: Implementa el bloque codificador con atención multi-cabeza
- **TransformerDecoder**: Implementa el bloque decodificador con atención causal

4.4 Hiperparámetros

Los hiperparámetros utilizados fueron:

Parámetro	Valor
Dimensión de embedding	256
Dimensión de capa densa	2048
Número de cabezas de atención	8
Tamaño del vocabulario	15,000
Tamaño de batch	64

Table 1: Hiperparámetros del modelo

4.5 Entrenamiento

El modelo base se entrenó durante 11 épocas (detenido por early stopping) utilizando:

- Optimizador: RMSprop
- Función de pérdida: Entropía cruzada categórica dispersa
- Guardado del mejor modelo según pérdida de validación

4.6 Mejora con GloVe

Para la versión mejorada:

- Se cargaron embeddings pre-entrenados GloVe de 300 dimensiones
- Se crearon matrices de embedding inicializadas con vectores GloVe
- Se realizó fine-tuning durante 15 épocas

5 Resultados y Discusión

5.1 Modelo Base

El modelo base alcanzó:

- Precisión en entrenamiento: 25.09%
- Precisión en validación: 23.10%
- Puntuación BLEU: 0.986

Época	Pérdida Train	Pérdida Val
1	5.075	2.854
2	2.904	2.441
3	2.493	2.264
5	2.165	2.215
7	2.025	2.204
10	1.916	2.285
11	1.893	2.296

Figure 1: Resumen de resultados del modelo base

5.2 Modelo Mejorado con GloVe

El modelo con embeddings GloVe mostró:

- Cobertura del vocabulario: 97.33% para inglés, 31.20% para español
- Precisión en entrenamiento al final: 24.81%
- Precisión en validación: 23.02%

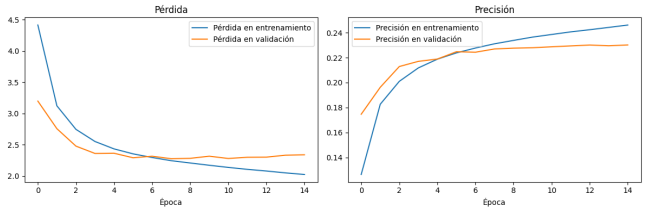

Figure 2: Progreso del entrenamiento con fine-tuning usando embeddings GloVe.

6 Dificultades y Limitaciones

6.1 Desafíos Técnicos

Durante el desarrollo se enfrentaron varios desafíos:

- **Transferencia de pesos:** Dificultades al transferir pesos del modelo pre-entrenado, con solo un pequeño número de capas con nombres idénticos
- **Compatibilidad de dimensiones:** Los embeddings GloVe (300d) tuvieron que adaptarse al modelo (256d)
- **Cobertura limitada:** Baja cobertura de vocabulario español en GloVe (31.20%)

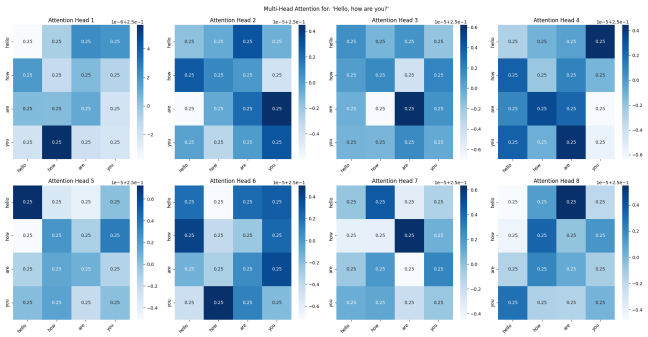

Figure 3: Heatmap de attention.

6.2 Limitaciones del Modelo

- **Secuencias cortas:** Límite de 20 tokens restringe la traducción de oraciones complejas
- **Vocabulario restringido:** 15,000 palabras resulta insuficiente para cobertura completa
- **Falta de contexto:** El modelo traduce oraciones aisladas sin considerar el contexto más amplio

7 Conclusiones

Este proyecto pudimos observar la compelejidad del desarrollo de sistemas de traducción automática. A pesar de los avances en arquitecturas como Transformer, la creación de traductores efectivos sigue presentando retos importantes:

- La necesidad de grandes cantidades de datos paralelos de calidad
- La representación adecuada de palabras en diferentes idiomas
- El manejo de diferencias estructurales y culturales entre idiomas
- Los requerimientos computacionales para entrenar modelos competitivos

En modelos lideres en NLP como BERT y sus derivados el numero de parametros a entrenar llega a ser 2 a 3 veces mas que los parametros usados en el presentre trabajo. Lo cual reafirma que las cantidades de datos y parametros necesarios para realizar un modelo NLP state-of-art.

Aunque los resultados obtenidos son modestos, el proyecto demuestra los fundamentos de la traducción neuronal moderna y las consideraciones prácticas en su implementación.