Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencias de la Computación

IIC3697 — Aprendizaje Profundo — 1' 2021

Informe de Avance Proyecto

Integrantes:

- Benjamín Farías
- Juan Hernández
- Benjamín Lepe

Introducción

Hasta hace poco, los humanos eran los únicos capaces de leer un texto, comprenderlo y luego responder preguntas relacionadas a este, sin embargo, esto ha cambiado considerablemente gracias a los avances en materias de procesamiento del lenguaje natural. Con técnicas de aprendizaje profundo se han logrado entrenar máquinas para que sean capaces de recibir un texto y luego responder preguntas sobre el contenido de dicho texto. Para lograr esto, se debe entrenar una red neuronal con cientos de miles de textos, los que traen consigo preguntas y respuestas que pueden o no tener relación con el contexto, como es el caso del Dataset presentado a continuación.

SQuAD Dataset

Para poder apoyar en el desarrollo de este campo de la inteligencia artificial, la universidad de Stanford creó el dataset **SQuAD (Stanford Question Answering Dataset).** Este dataset contiene una gran cantidad de textos que tratan distintos temas, donde cada uno además trae un **set de preguntas junto a sus respectivas respuestas.** Por otro lado, también incluye algunas preguntas que no tienen respuesta bajo el contexto dado, y por lo tanto actúan como distractores. La idea de este dataset es que los modelos logren responder correctamente a preguntas dado un contexto previo (en forma de texto), y que además se abstenga de responder cuando la pregunta no tiene respuesta.

Estado del Arte

Al revisar el ranking de **SQuAD competition**, nos damos cuenta que hoy en día existe una gran cantidad de métodos que permiten obtener predicciones exactas con una **precisión mayor al 86 %**, lo que supera al rendimiento humano. Una gran cantidad de estos métodos son mejoras o ensambles de **BERT (Bidirectional Encoder Representations from Transformers)**, una técnica desarrollada por Google para el pre-entrenamiento del procesamiento del lenguaje natural.

Entre las mejoras y ensambles más destacados de BERT se encuentran los siguientes:

- **ALBERT (A Lite BERT):** Al igual que el algoritmo original, este fue desarrollado por el equipo de Google AI. Consiste en una variación *lightweight* de BERT. Esta no sólo ha demostrado contar con un número menor de parámetros que el algoritmo original, sino que además es capaz de obtener mejores resultados en SQuAD 2.0.
- **RoBERTa:** Fue implementado por Facebook AI. Consiste en una versión de BERT entrenada por un mayor tiempo sobre un número más grande de ejemplos y con diferentes hiperparámetros. Específicamente, se incrementó el *learning rate* y se usaron *mini-batches* más grandes. Además, se eliminó el objetivo de la siguiente oración dentro del pre-entrenamiento de BERT.
- **BigBird:** Fue implementado por Google AI y es una mejora de BERT aún más reciente que ALBERT (2020). Debido a su mecanismo de *full self-attention*, BERT posee un crecimiento cuadrático de requerimientos de memoria por cada input token. BigBird propone un mecanismo de atención dispersa, la cual disminuye el uso de recursos y permite procesar inputs 8 veces más grandes que BERT.

Posibles Experimentos

Tenemos pensado explorar las siguientes vertientes para el desarrollo de este proyecto, **utilizando técnicas basadas en BERT:**

- **Mejoras:** Probar agregando, eliminando o modificando ciertas partes de las técnicas, intentando encontrar una arista por la que se pueda mejorar el rendimiento en SQuAD. Esto podría significar el agregar/quitar capas a los modelos, o cambiar escalares que controlen el aprendizaje, entre otros hiperparámetros. Otra opción puede ser probar con distintos clasificadores, tal que puedan aumentar la eficiencia al momento de entregar una respuesta.
- **Ensambles:** Combinar los outputs de distintas técnicas populares, esperando generar un ensamble que obtenga la información de todas estas, y sea capaz de decidir entre los output o combinar las respuestas bajo algún criterio específico. De esta forma se tendría un *approach* más robusto que simplemente usar las técnicas por separado.
- **Combinación entre Mejoras y Ensambles:** Combinar técnicas entre sí, pero a nivel de la implementación interna de cada una, es decir, generar ensambles que además incluyan mejoras a los modelos individuales.

Referencias

- **SQuAD Competition:**
  https://rajpurkar.github.io/SQuAD-explorer/
- **Hugging Face Transformers:**
  https://huggingface.co/transformers/
- **BERT Explained:**
  https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-nlp-f8b21a9b6270
- **ALBERT Implementation:**
  https://github.com/google-research/albert
- **RoBERTa Implementation:**
  https://colab.research.google.com/github/pytorch/pytorch.github.io/blob/master/assets/hub/pytorch_fairseq_roberta.ipynb