College Web-Application Large Language Model

Noah Sturgill (nsturgill2001@gmail.com)
The University of Virginia's College at Wise

Abstract

This work presents a Large Language Model (LLM) based on the transformer architecture, implemented using PyTorch. The model shall operate as an embedded system in a web application tailored to answer questions for students at the University of Virginia's College at Wise. The proposed model utilizes the GPT-2 tokenizer and is trained on a custom dataset prepared from diverse text sources relevant to the college. The implemented transformer architecture includes features such as positional encoding, enabling the model to capture contextual information and dependencies in sequential data. The training pipeline utilizes data loading, tokenization, and model training with a focus on optimizing the model for efficient GPU usage. The model architecture comprises an embedding layer, Transformer encoding layers, and a linear decoder. Additionally, the model incorporates a mechanism to dynamically generate masks for self-attention, enhancing its ability to capture sequential dependencies. The model's training process involves minimizing the cross-entropy loss using the Adam optimizer. The training and validation status is monitored with informative status messages, displaying epoch-wise training and validation losses. The training loop is equipped with gradient clipping to mitigate exploding gradients. The research conducted on this model may showcase the feasibility of catered LLMs across educational institutions.

2 Introduction

In the rapidly evolving landscape of natural language processing (NLP), the deployment of advanced language models has become instrumental in addressing diverse challenges. At the University of Virginia's College at Wise (UVA Wise), there exists a unique informational ecosystem characterized by specific needs and challenges that standard language models may not address effectively. The institution seeks innovative solutions catering to the distinct requirements of its community, emphasizing the need for a tailored and intelligent support system. This system shall seamlessly integrate into web applications, providing users with precise, contextually relevant information about the college. Furthermore, the complexity of the college's community needs necessitates a model capable of understanding and generating human-like text, ensuring natural and effective interaction. The implementation best suited for capturing contextual language to allow for complex queries is the recently pioneered transformer architecture (Vaswani et al., 2017; Devlin et al., 2018).

The introduction of transformer architecture has revolutionized the field of NLP, offering unprecedented capabilities in handling sequential data through mechanisms such as positional encoding and self-attention. These features enable models to capture contextual information over long sequences, thereby significantly enhancing the quality of generated text and the understanding of user queries (Vaswani et al., 2017). Despite these advancements, the application of transformer-based Large Language Models (LLMs) within the academic support context, especially in smaller institutions such as UVA Wise, remains unexplored.

Current implementations of LLMs in educational settings primarily focus on general-purpose applications, lacking the customization necessary for specific institutional contexts (Karsenti, 2019). Consequently, there is a pressing need to develop and deploy large language models that are not only grounded in cutting-edge transformer architecture but also fine-tuned with data reflective of unique informational

ecosystems. Such development efforts would bridge the gap between NLP technologies and the educational landscape; closing the gap assists in long-term student success (Karsenti, 2019).

This work aims to address the outlined gap by presenting a transformer based LLM, implemented using PyTorch and tailored to the specific needs of UVA Wise. Through careful training on a custom dataset compiled from diverse text sources relevant to the college, the model is expected to provide accurate and contextual answers to user queries. Such interactions shall significantly enhance the academic support system's effectiveness.

3 Methodology

This section elucidates the methodology employed in designing, implementing, and training a Large Language Model (LLM) tailored to serve the University of Virginia's College at Wise. The model leverages the transformer architecture, renowned for its proficiency in handling sequential data and capturing long-range dependencies. Such dependencies are crucial for understanding and generating natural language accurately (Vaswani et al., 2017).

3.1 Model Architecture

The architecture is instantiated utilizing PyTorch and comprises an embedding layer, multiple Transformer encoding layers, and a linear decoder for output generation. The model is distinguished by its integration of a Positional Encoding module and a dynamic masking mechanism for self-attention. This mechanism facilitates the comprehension of sequential dependencies and contextual nuances.

$$PE(pos, 2i) = sin\left(pos/10000^{2i/d_{model}}\right)$$
$$PE(pos, 2i+1) = \cos\left(pos/10000^{2i/d_{model}}\right)$$

The preceding equations demonstrate the calculation of positional encoding. In the equation, pos is the position, *i* is the dimension, and d_{model} is the model's dimension size. This equation implements the sinusoidal function used to compute the positional encoding (Vaswani et al., 2017).

3.2 Data Preparation and Training Procedure

The methodology for creating this model utilizes the creation of a custom dataset by aggregating text sources pertinent to UVA Wise. Utilizing the GPT2-tokenizer, texts are tokenized and structured into a dataset optimized for training and validation. The dataset is dynamically partitioned, with 90% allocated for training and 10% for validation. This dynamic segregation ensures appropriate exposer to linguistic structures.

Training leverages the Adam optimizer, focusing on minimizing the cross-entropy loss to effectively adjust model parameters; the Adam optimizer outperforms similar optimization algorithms in large datasets (E.M. Dogo et al., 2018). Notably, gradient clipping is employed to mitigate the issue of exploding gradients, a phenomenon prevalent in deep neural networks.

3.3 Evaluation and Generation

$$L = -\sum y_{o,c} \log\left(p_{o,c}\right)$$

The model is evaluated using the cross-entropy loss measured on the validation dataset. Additionally, the model's ability to generate contextually coherent and relevant responses is tested through a generation function which dynamically adapts the temperature of the softmax probabilities to influence the diversity of the generated responses. The preceding equation showcases the cross-entropy loss equation where $y_{o,c}$ is a binary indicator of whether class label *c* is the correct object for observation *o*, and $p_{o,c}$ is the predicted probability of observation *o* being of *c* class.

4 Versions

The model is being developed using Python 3.11.4, CUDA compilation tools version 11.3 (Ubuntu 22.04.2), CUDA Version 12.3 (Ubuntu 22.04.2), NVIDIA GeForce RTX 3070Ti Laptop GPU, and WSL2 (Windows 11 Home).

5 Conclusion

Preliminary development of the model is promising. The training phase of the model is the next phase of development. A search for similar models is being conducted to establish a performance benchmark. Future iterations of the model will be evaluated on accuracy, precision, and efficiency.

6 Limitations

The limitations of the research being conducted derive from the limited resources in the development of the model. Computing power to train the model is currently limited to a singular workstation; the code is currently being written by a singular individual. Other limitations include the scarcity of similar models limiting the performance evaluation of the model.

References

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems. (NIPS'17). Curran Associates Inc., Red Hook, NY, USA, 6000–6010.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Thierry Karsenti. 2019. Artificial intelligence in education: The urgent need to prepare teachers for tomorrow's schools [Chronique]. *Formation et profession,* *27*(1), 105. https://dx.doi.org/10.18162/fp.2019.a166

E. M. Dogo, O. J. Afolabi, N. I. Nwulu, B. Twala, and C. O. Aigbavboa. 2018. A Comparative Analysis of Gradient Descent-Based Optimization Algorithms on Convolutional Neural Networks. In 2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS), 92–99. DOI:https://doi.org/10.1109/CTEMS.2018.8769211