Expanding the knowledge of deep neural networks

Adrienn Molnár

Introduction. The primary goal of this semester's project was to achieve a comprehensive understanding of the expansion of knowledge in large language models (LLMs), a vital and relevant issue in today's context, especially in business environments where handling confidential data is crucial. Companies aim to utilize LLMs for searching and interacting with their proprietary data, while ensuring that this information remains undisclosed and secure.

Transformer. Transformer [Vaswani et al., 2017] models have revolutionized natural language processing and machine learning by introducing an architecture based on the attention mechanism, diverging from previous reliance on recurrent or convolutional neural networks.
Transformers decompose their inputs into smaller units, known as tokens, which could be letters in language or segments in vision tasks. This tokenization is a critical step in processing and understanding the inputs.
Let the transformer language model be represented by the function f. The input for f is a series of tokens, $(x_1, x_2, ..., x_n)$, where each x_i is an input token transformed into a high-dimensional vector using an embedding algorithm.
The self-attention module is a central element in such learning systems. In the context of transformer models, each input token is processed to generate three distinct matrices: Query (Q), Key (K), and Value (V). These matrices are created through learned linear transformations applied to the input embeddings. Specifically, for a given input token indexed by i, the respective matrices are represented as Q_i, K_i, and V_i.
The self-attention mechanism operates by assessing the relationships between different tokens. It essentially directs the 'focus' or 'attention' of the model to different parts of the input sequence, allowing the model to weigh the importance of each token relative to others in the sequence. For a single token, this operation can be formalized as follows:

$$\text{Attention}(Q_i, K_i, V_i) = \text{softmax}\left(\frac{Q_i K_i^T}{\sqrt{d_k}}\right) V_i$$

The term $\sqrt{d_k}$ serves as a scaling factor, where d_k represents the dimension of the Queries and Keys. This scaling is crucial to prevent the softmax function from entering regions with extremely small gradients during training, thereby stabilizing the learning process.

Large language models. A **language model** is a probabilistic model that assigns a probability to a sequence of n words by learning from a corpus of text data. Formally, a language model estimates the probability distribution $P(W)$ over sequences of words W, where $W = w_1, w_2, \ldots, w_n$ represents a sequence of words. The probability of a word sequence is given by:

$$P(W) = P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i | w_1, w_2, \ldots, w_{i-1})$$

where $P(w_i | w_1, w_2, \ldots, w_{i-1})$ is the conditional probability of word w_i given the preceding words in the sequence.
In language modeling tasks, transformers have demonstrated exceptional performance, effectively predicting and generating text sequences. Their architecture has become foundational in the field, making them integral to the advancement of large language models in natural language processing.

Fine-Tuning Transformer Models. Fine-tuning involves adjusting the parameters of a pre-trained model for a specific task. Let $D = \{(x_i, y_i)\}_{i=1}^{N}$ be a dataset for the new task, where x_i is an input and y_i is the corresponding label. The model is fine-tuned using a loss function $\mathcal{L}$ and stochastic gradient descent (SGD). The loss function measures the error between the model's prediction $f(x_i)$ and the true label y_i for each pair in D. The objective is to find parameters Θ that minimize the loss:

$$\Theta^* = \arg\min_{\Theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(f(x_i; \Theta), y_i),$$

where Θ^* represents the optimized model parameters.

In-context Learning. In-context learning Dong et al. [2022] refers to the ability of large language models to adapt to new tasks without explicit retraining or fine-tuning. This is achieved by providing a context or a series of examples within the input itself. The model leverages patterns seen during training to infer the task and generate appropriate responses. Mathematically, given a context C and a task T, the model predicts an output O based on the learned representation:

$$O = f(C, T; \Theta),$$

where Θ are the parameters of the model.

Retrieval-Augmented Models. Retrieval-augmented models (RAG) [Lewis et al., 2020] bring a transformative approach to natural language processing by combining transformer architectures with an external data retrieval system. Functionally represented as f, these models enhance the input x with relevant information R extracted from an external dataset D. This approach significantly improves their processing and generation abilities. The model's output y is the result of integrating the input with the externally retrieved data:

$$y = f(x, R(x; D); \Theta)$$

In this framework, $R(x; D)$ acts as the retrieval function that selects relevant information based on x, and Θ are the parameters of the model. Such integration provides the model access to a vast array of information beyond its original training data, enhancing its overall performance and capabilities.

However, a key advantage of RAG models becomes evident when compared to fine-tuned models. While fine-tuning approaches adjust a model's parameters to specific tasks, they often lack clarity in how and why a particular response is generated. In contrast, RAG models offer a level of transparency that is crucial for certain applications: the origin of a response can be directly traced back to the retrieved documents.

Conclusion and future work. During this project, we explored transformer models and their applications in natural language processing, highlighting the innovations and challenges in fine-tuning and retrieval-augmented methods. We identified the need for greater transparency and efficiency in these models.
In the upcoming semester, I plan to focus on the evolving field of RAG models, particularly investigating document organization and text segmentation methods. For example, exploring ways to segment text into document units or the use of different embedding techniques could provide insights into improving these models' effectiveness and interpretability.

References

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. A survey for in-context learning. *arXiv preprint arXiv:2301.00234*, 2022.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.