Strategies for good implementation in fine-tuning models.

1. Strategy

To address the correct way to use a pre-trained model, it is essential to identify some strategies to help you succeed now in fine-tuning an LLM. It is necessary to mention that those strategies apply to Classification and Generative text models. During the search process, I could identify the following strategies and improve Thomas' exercise.

1.1. Check if fine-tuning is the best solution.

If you are exploring fine-tuning, OpenAI recommends first attempting to get good results with prompt engineering, prompt chaining, and function calling. These are some cases if you want to fine-tune and you can improve outcomes:

- Setting the style, tone, format, or other qualitative aspects.
- Improving reliability in producing a desired output.
- Correcting failures to follow complex prompts.
- Handling many edge cases in specific ways.
- Performing a new skill or task that's hard to articulate in a prompt.

1.2. Find the dataset.

Before fine-tuning a pre-trained model, you must download or build your dataset. Hugging Face has a variety of datasets, especially if you want to explore how the model works. Still, if you have a specific objective for a unique task, the best is to build your dataset or generate a dataset through GPT or Llama models.

1.2.1. Using generative data.

First, you must find the best prompt that gives the answer you want. That means you must optimize your prompt. Then, you can create diverse demonstration conversations like the optimized prompt. There are six strategies that OpenAI recommends:

- Write clear instructions.
- Provide reference text.
- Split complex tasks into simpler subtasks.
- Give GPTS time to "think".
- Use external tools.
- Test changes systematically

1.3. Fine-tune a pre-trained model.

In this step, you must pick the best framework you want to use. In the case of HF, you can use Pytorch and TensorFlow. Or, if you want to use generative text such as GPT, you can use gpt-3.5-

turbo, babbage-002, or davinci-002. Also, make sure to understand the different parameters that the models can have to modify or optimize and get the best performance of your model.

1.4. Use your fine-tuned model.

Make sure that your fine-tuned model has good quality testing metrics such as training loss, training token accuracy, test loss, and test token accuracy. The previous statistics give you an idea of how good your model is.

2. References

1. Hugging Face. Fine-tune a pretrained model. https://huggingface.co/docs/transformers/training.
2. OpenAI. Fine-tuning. https://platform.openai.com/docs/guides/fine-tuning.
3. Kourosh Hakhamaneshi & Rehaan Ahmad. (2023, August 11). https://www.anyscale.com/blog/fine-tuning-llama-2-a-comprehensive-case-study-for-tailoring-models-to-unique-applications.
4. OpenAI. GPT best practices. https://platform.openai.com/docs/guides/gpt-best-practices/six-strategies-for-getting-better-results.