Fine-Tuning Large Language Models Using Self-Generated Labeled Data

2310089 Yusuke Takamori

In recent years, large language models (LLMs) have demonstrated remarkable performance across a wide range of natural language processing tasks, including natural language understanding and generation. In general, the application of LLMs follows a two-stage procedure consisting of pre-training on large-scale corpora and subsequent fine-tuning to adapt the model to specific downstream tasks. Along with this progress, parameter-efficient fine-tuning methods have been proposed, enabling the adaptation of LLMs to downstream tasks even in environments with limited computational resources, thereby further facilitating the practical use of LLMs. Nevertheless, a fundamental challenge remains, as fine-tuning typically requires a large amount of labeled data. In supervised fine-tuning, each downstream task demands a labeled dataset of sufficient quantity and quality, the construction of which incurs substantial human and time costs. This issue is particularly pronounced in highly specialized domains such as medicine, law, and finance, where large-scale public datasets are sparse and annotation often requires expert involvement, making data collection itself a major bottleneck.

Against this backdrop, self-generated data-based approaches, in which LLMs generate training data for their own learning, have recently attracted increasing attention. Previous studies, including Self-Instruct, have shown that using pseudo-labeled data generated by the model itself can improve the performance of a downstream task without relying on human annotation. However, much of the existing work has primarily focused on the design of data generation processes, while comparatively little attention has been paid to how individual generated samples should be evaluated and selected after generation, or how different selection strategies influence the performance of downstream tasks. In particular, systematic investigations comparing the effects of data quality assessment and filtering methods for self-generated data across multiple downstream tasks remain limited.

The goal of this study is to clarify under what conditions fine-tuning with self-generated labeled data produced by LLMs is effective, as well as to identify its inherent limitations. In particular, we focus on the quality assessment and selection of generated samples after data generation, and systematically analyze how different filtering strategies, such as generation probability filtering and LLM-as-a-judge filtering, affect both the quality of training data and the downstream task performance of LLMs.

The proposed method consists of three stages: (1) generation of labeled samples, (2) filtering of self-generated samples, and (3) fine-tuning using the

filtered samples. First, the LLM is prompted with a natural language description of the target downstream task, and generates pseudo-labeled samples that satisfy the the task-specific requirement of a pair of an input and output. To ensure the diversity of generated samples, different keywords are provided in the prompts, instructing the model to generate samples related to each keyword.

Next, confidence-based filtering strategies are applied to remove samples containing incorrect labels, semantic inconsistencies, or redundant information. Depending on characteristics of the task, three types of filtering methods are employed: (1) similarity filtering, which evaluates the semantic correspondence between two texts within a sample, (2) generation probability filtering, which relies on the output's generation probability predicted by the LLM, and (3) LLM-as-a-judge filtering, in which the LLM itself assesses the validity of each sample with respect to the task definition. These methods evaluate sample quality from complementary perspectives, including semantic consistency, textual fluency, and relevance to the task.

Finally, the filtered self-generated data are used to fine-tune the LLM using LoRA. By adopting this parameter-efficient fine-tuning approach, our method enables effective adaptation to downstream tasks under limited computational resources. The proposed framework is applicable to a wide range of downstream tasks, including both classification and generation tasks.

To evaluate the effectiveness of the proposed method, experiments on multiple downstream tasks, including both classification and generation tasks, were conducted. Specifically, three classification tasks were considered, namely Recognizing Textual Entailment, Sentiment Analysis, and Natural Language Inference in a legal-domain. In addition, a text generation from structured data, called End-to-End Natural Language Generation (E2E NLG), was considered as a generation task. A zero-shot inference with a pre-trained LLM was used as the baseline, and compared with inference using LLMs fine-tuned on self-generated labeled data, i.e., our proposed methods. For the latter, we evaluated fine-tuning without filtering and fine-tuning with three filtering strategies: similarity filtering, generation probability filtering, and LLM-as-a-judge filtering. Across all tasks, the same LLM (Llama-3) was used consistently for sample generation, quality evaluation, and fine-tuning.

Experimental results showed that fine-tuning with self-generated data consistently improved the performance over the baseline across all tasks. Moreover, in many cases, applying filtering to self-generated samples further enhanced the performance of the downstream tasks. For example, in the Sentiment Analysis task, the accuracy of the pre-trained model was 0.54. The accuracy was increased to 0.82 by fine-tuning the LLM using the self-generated labeled dataset without filtering, and further improved to 0.91

when LLM-as-a-judge filtering was applied. For the generation task, the proposed method achieved an improvement of 0.036 in BERTScore F1 over the baseline.

A comparison of filtering strategies revealed that LLM-as-a-judge filtering was the most effective for tasks with relatively explicit input-output correspondences, such as Sentiment Analysis and E2E NLG. In contrast, for tasks requiring sentence-level reasoning or domain-specific knowledge, including Recognizing Textual Entailment and legal-domain NLI, similarity-based filtering yielded better performance, while the effectiveness of LLM-as-a-judge filtering and generation probability filtering was limited. These results indicated that the quality of self-generated data has a substantial impact on the performance of downstream tasks, and that the optimal filtering strategy is highly dependent on the characteristics of the task.