GPT-4o Machine Learning Fine-Tuning: quality control & data filtering

Overview

This work is part of the LLM fine tuning I did to make an agent that could communicate with customers over live chat based on recorded human examples. In this section I've described the filtering process that went into curating the training data for the deployed model, including the problems with the dataset to motivate these choices.

The problem

Much of the data was over the Covid period, meaning many chats were obsolete or contained out-of-date company policy. Some chats were very short (only one conversation turn) or were malformed. Employee names and customer names needed to be anonymised.

Some chats contained customers swearing.

Some chats represented poor training examples, in more nuanced ways that couldn't be filtered out through a keyword or keyphrase filter.

What was achieved?

I was able to reduce the occurrence of poor training data to produce a useful fine-tuned LLM. This was evaluated by the head of sales at the company, and was shown to be effective after integration into the company live chat software on real live chats. The evaluation covered the bot's jewellery knowledge, its ability to engage to keep the conversation going, and finally its ability to naturally get the customer to provide their email or phone number so other sales employees could continue the sales process over another channel with a higher conversion rate.

The head of sales judged the model's conversational performance, knowledge, and sales technique to be comparable to those of a trained sales employee and therefore judged the model suitable for deployment.

Filtering rules

I selected high-performing sales employees, whose historical live chat conversations would be considered for use in the training data. This puts a large focus on the best live chat examples when training.

I removed Covid-influenced training examples through date-based filtering.

I removed incorrect policy training examples using date-based filtering, keyword and keyphrase matching and a final manual review of remaining conversations.

I required each live chat conversation to contain at least two customer messages and at least one employee response. Live chats were always started by the customer, and were subsequently responded to except when the message was obviously intended to be hurtful or hateful. I couldn't train the model on any result where there was no recorded employee response.

Further, I decided not to train on any example where the customer didn't reply because it implies the response may not have been optimal or engaging enough. This decision lessened the amount of training data but only slightly. The variety of first responses that we wanted the model to provide eloquently was small, and our training data for successful conversation openers was already well covered by successful chats.

Programmatic filtering was not enough to clean training data

After I removed many live chats that depended on training the AI on time-dependent company policy, the resultant fine-tuned large language model was not trained well enough to deploy. The issue was poor training examples still polluting the training set. The remaining problems concerned the quality of the responses and the judgement demonstrated in the conversation, making them difficult to identify through simple keyword or structural checks.

I solved this problem through a final stage of manual dataset filtering. After consulting the head of sales and reviewing the company's sales training documentation, I assessed the remaining transcripts against the company's expectations for an effective customer conversation. Over three working days, I removed examples that demonstrated poor sales judgement or unsuitable conversational behaviour, selecting training examples that reflected how I wanted the model to respond.

Evaluation and outcome

The head of sales initially reviewed the fine-tuned model and judged its responses to be below the standard of trained sales employees, making it unsuitable for deployment. After I manually curated the training data and retrained the model, he judged its conversational performance to be good relative to that of a trained sales employee. The model already demonstrated general knowledge of diamonds and jewellery before retraining and retained that knowledge afterwards. Company-specific product information and policies were introduced later through a vector database and were outside the scope of this evaluation. We also compared responses at several inference temperature settings.

Examples of changes to the training data to improve the model

One problem was losing the timing between messages. In one conversation, a customer was looking for a princess-cut engagement ring but had not found something they would be

comfortable buying. The employee suggested three designs. In the unfiltered training data, the end of the employee's response read:

Before — unfiltered training response

> These are all princess cut with high settings Hi are you still there?

The original email transcript shows that these two messages were sent minutes apart, except the training data processing interpreted them as one employee response. Grouping consecutive employee messages into one turn removed the distinction.

After — amended training response

> These are all princess cuts with high settings. If you let me know what you like/don't like I can send you more designs in line with what you might like.

This was preferred by the sales team because it gives the customer a specific next step and improves customer engagement.

Other replies depended on information missing from the training input. Another customer said they were just browsing and, when asked, confirmed that they had no particular deadline. The employee then replied:

Before — original reply

> You are currently looking at the Magnolia

The customer had not named the Magnolia anywhere in the recorded conversation. The employee may have been able to see the visitor's current product page, but that information was absent from the training input.

After — amended training reply

> Is there a design you are looking at? If you let me know I could give you some similar designs you may like.

The reply now asks for the missing information and explains how providing it would help.

Some conversations also mixed useful dialogue with temporary company procedures. A third conversation concerned an upcoming showroom visit. The employee answered the customer's question about facilities, then gave pandemic-specific instructions about sanitiser, masks and symptom screening.

Before — excerpt from the later employee reply

> We also have complimentary disposable masks should you like to wear one (not obligatory). May I ask if either of you have experienced any fever, muscle aches, persistent dry cough or loss of taste/ smell in the last 7 days?

I retained the answer about showroom facilities and removed the screening exchange. The amended conversation closes with:

After — closing reply in the amended conversation

> Wonderful, many thanks! See you tomorrow.

The revised versions of all three examples were retained in the final saved training dataset.

I also amended conversations that were already well written to give the customer a clearer reason to continue. In this example, the employee established the customer's preferences and recommended a ring design. The original response ended with a general invitation to give feedback.

Before — excerpt from the later employee reply

> "Have a look at this one and let me know what you think. "

After — closing reply in the amended conversation

> Have a look at this one and let me know what you think. If you could let me know what you like/dislike about this design, then I could get back to you with other examples you might like

I expanded that invitation to ask what the customer liked or disliked about the design and explain that their answer would help identify further options.