ДОДАТОК А

Перелік джерел посилання за науковими напрямами керівника та науковців кафедри Програмної Інженерії

5. K. Smelyakov, A. Chupryna, D. Sandrkin and M. Kolisnyk, "Search by Image Engine for Big Data Warehouse," 2020 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream), Vilnius, Lithuania, 2020, pp. 1-4, doi: 10.1109/eStream50540.2020.9108782

6. K. Smelyakov, A. Chupryna, O. Bohomolov and I. Ruban, "The Neural Network Technologies Effectiveness for Face Detection," 2020 IEEE Third International Conference on Data Stream Mining & Processing (DSMP), 2020, pp. 201-205, doi: 10.1109/DSMP47368.2020.9204049.

ДОДАТОК Б

ЗВІТ РЕЗУЛЬТАТАМИ ПЕРЕВІРКИ НА УНІКАЛЬНІСТЬ ТЕКСТУ В БАЗІ ХНУРЕ

UNICHECK
by Turnitin

Ім'я користувача:
Кардаш Євген Вікторович каф.ПІ

Дата перевірки:
13.06.2024 11:47:40 EEST

Дата звіту:
13.06.2024 11:57:14 EEST

ID перевірки:
1016356244

Тип перевірки:
Doc vs Internet + Library

ID користувача:
100013622

Назва документа: 2024_М_ПІ_ІПЗм-22-1_Кіпріч_І_С_скорочений

Кількість сторінок: 34 Кількість слів: 6224 Кількість символів: 45779 Розмір файлу: 440.81 KB ID файлу: 1016160416

2.44% Схожість

Найбільша схожість: 1.48% з Інтернет-джерелом (https://www.oksim.ua/2023/07/20/vstup-do-velykyh-movnyh-modelej-..

2.41% Джерела з Інтернету 41 ..... Сторінка 36

0.32% Джерела з Бібліотеки 33 ..... Сторінка 36

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи 16

ДОДАТОК В

Тези доповіді на конференції «ІНФОРМАЦІЙНІ ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ» XXVIII міжнародного молодіжного форуму «РАДІОЕЛЕКТРОНІКА ТА МОЛОДЬ У XXI СТОЛІТТІ»

UDC 004.8:519.816 DOI: https://doi.org/10.30837/IYF.IIS.2024.510

RESEARCH ON THE INTEGRATION OF AI FINE TUNED MODELS IN CUSTOMER SUPPORT SYSTEMS

Kiprich Ivan
ScD, Professor, Software Engineering Department, Smelyakov Kirill
Kharkiv National University of Radio Electronics, dep. SE.,
Kharkiv, Ukraine
e-mail: ivan.kiprich@nure.ua

The subject of this work is dedicated to studying fine-tuning techniques of large language models within the context of custom domain. The goal of the work is to analyze methods of fine-tuning large language models, compare their productivity and quality of results with limited amount of training data, and explore their utilization in customer support systems. This article describes preparing a dataset, various methods of fine-tuning and evaluates them using a custom data set. Domain specific help articles and the client queries in a "question-answer" format were used as data.

Introduction

Customer support systems play a crucial role in the product lifecycle and help shape and increase customer loyalty to the product. There are several types of client inquiries: informational requests, technical support, feedback, and various scenarios for their processing.

During the processing of informational requests, various difficulties may arise related to vague question formulation, unavailability of information, large knowledge base volume, answer relevance, data privacy, and security. The use of large language models (LLMs) can reduce the processing time of customer queries and provide answers based on the company's knowledge base.There are several types of large language models for working with natural texts such as transformers, recurrent neural networks (RNN), and models based on convolutional neural networks (CNN). Smelyakov [1] evaluated the performance of some modern convolutional neural networks. In the domain of image search and retrieval, Smelyakov et al. (2020) [2] introduced an innovative approach that can be partially applicable for natural language processing tasks.

Transformer architecture was introduced by Vaswani et 2017 [3], which has become the basis for many state-of-the-art natural language processing models, including BERT, GPT, and others. The main concepts of Transformer is tokenization, embedding, attention, pretraining and transfer learning. In the context of LLM, transfer learning involves fine-tuning a pretrained model on a smaller, task-specific dataset to achieve high performance in solving a new task.

There are several types of parameter tuning for large language models for use in a particular field. Full parameter tuning optimizes the entire model for the target domain, but require large amount of labeled data and computationally expensive. Parameter-efficient fine tuning often involve selectively updating a

Рисунок В.1 – Тези доповіді (рисунок виконаний самостійно)

subset of the model parameters while keeping the rest fixed or updating them with a smaller learning rate, particularly useful in scenarios where labeled data is scarce or when deploying models to resource-constrained environments.

Low-Rank Adaptation of Large Language Models or LoRA [4] freezes the pre-trained model weights and injects trainable rank decomposition matrices into each layer of the Transformer architecture. Q-LoRA starts with the idea of approximating the weight matrices of the pre-trained language model with low-rank matrices. Quantization is a compression technique that reduces the bit width of the parameters and/or activations of LLMs to improve their efficiency and scalability [5].

Experiment Planning

During the preparation process, the knowledge base of the Comsend project was exported, along with the history of requests and responses from the customer support system. Articles were saved in txt format without additional markup. In total, 139 articles were processed, with a total volume of 150,157 characters. For processing the data obtained from the request history, the "question-answer" format was chosen. Duplicate questions and questions without answers were filtered out. Personal data was replaced with placeholders. For comparison, we used open-source large language models with different fine-tuning parameters such as LLaMa-2-7B and GPT-J 6B.

The LLaMa series is recognized for its efficiency and the ability to perform a wide range of language tasks with less computational power compared to some other models.

GPT-J-6B is an open-source large language model (LLM) developed by EleutherAI in 2021. As the name suggests, it is a generative pre-trained transformer model designed to produce human-like text that continues from a prompt. The optional "6B" in the name refers to the fact that it has 6 billion parameters.

In research we trained LLaMa-2-7B and GPT-J-B6 with 4-bit quantization and compared it with the untuned LLaMa-2-7B model. We used a Python programming language in the PyCharm development environment.

To fine tune models was used Google Colab, a free cloud service based on Jupyter Notebook. After model fine tuning, an API was developed for integration into the customer support system. For new customer queries, three response options were generated by different models.

The support manager evaluated the relevance of each response on a scale from 1 to 5, where 1 represented the worst variant and 5 the best. 124 customer queries were processed, and the following results were obtained (tabl. 1).

Рисунок В.2 – Тези доповіді (рисунок виконаний самостійно)

Table 1 – Customer support answer rates

Model	Support management rates				
	1	2	3	4	5
GPT-J-B6 fine tuned	18	21	31	28	26
LLaMa-2-7B fine tuned	9	11	20	41	43
LLaMa-2-7B	28	31	29	23	13

The manager's rating of 1 or 2 was interpreted as negative, while 3, 4, 5 were interpreted as positive, resulting in the following values (tabl. 2).

Table 2 – Results of prompts

Model	Negative	Positive
GPT-J-B6 fine tuned	31,5%	68,5%
LLaMa-2-7B fine tuned	16,1%	83,9%
LLaMa-2-7B	47,6%	52,4%

Conclusion

Based on results of research, the following main conclusion can be made. Model trained on global dataset without fine tuning significantly fall behind models fine tuned with domain specific dataset. LLaMa-2-7B produces more relevant results, but takes more resources to fine tune. Providing detailed domain specific dataset for fine tuning increases response accuracy, but on other hand requires more computational power.

References:

1. K. Smelyakov, A. Chupryna, D. Sandrkin and M. Kolisnyk (2020). Search by Image Engine for Big Data Warehouse. IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream). 1-4, https://doi.org/10.1109/eStream50540.2020.9108782.

2. K. Smelyakov, A. Chupryna, O. Bohomolov and I. Ruban (2020). The Neural Network Technologies Effectiveness for Face Detection. IEEE Third International Conference on Data Stream Mining & Processing (DSMP). 201-205,https://doi.org/10.1109/DSMP47368.2020.9204049.

3. Hu, J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., & Chen, W. (2012). LoRA: Low-Rank Adaptation of Large Language Models. ArXiv, abs/2106.09685 URL: https://doi.org/10.48550/arXiv.2106.09685.

4. Xu, Y., Xie, L., Gu, X., Chen, X., Chang, H., Zhang, H., Chen, Z., Zhang, X., & Tian, Q (2023). QA-LoRA: Quantization-Aware Low-Rank Adaptation of Large Language Models. ArXiv, abs/2309.14717. URL: https://doi.org/10.48550/arXiv.2309.14717.

Рисунок В.3 – Тези доповіді (рисунок виконаний самостійно)

ДОДАТОК Г

ПРЕЗЕНТАЦІЙНИЙ МАТЕРІАЛ ДО КВАЛІФІКАЦІЙНОЇ РОБОТИ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

NURE ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Дослідження моделі інтеграції соціально-зорієнтованих інформаційних моделей штучного інтелекту для сервісів роботи з клієнтами

Кіпріч Іван Сергійович, ІПЗм-22-1

Науковий керівник: проф кафедри ПІ
Смеляков Кирило Сергійович

25 червня 2024

software engineering

Дослідження

Великі мовні моделі (LLMs) можуть застосовуватися для вирішення широкого спектру завдань в області обробки природної мови, наприклад:

- Генерація текстів (чат боти та віртуальні асистенти)
- Питання-відповіді (пошукові системи, служби підтримки клієнтів)
- Аналіз настроїв (моніторинг бренду, маркетингові дослідження)
- Класифікація текстів (спам фільтри, категоризація контенту)
- Переклад

Метою роботи є оцінка можливості використання постнавчених великих мовних моделей в системах підтримки користувачів, та аналіз методів постнавчаня

software engineering

Постановка задачі

Великі мовні моделі навчені на загальних даних. Застосування таких моделей без додаткових налаштувань в певній галузі не дає задовільних результатів. Під час роботи необхідно дослідити методи постнавчання та можливість навчання різних моделей з застосуванням даних користувачів.

Очікується що постнавчені моделі будуть генерувати текст відповідно до запиту, використовуючи дані та інформацію специфічну для домена.

software engineering

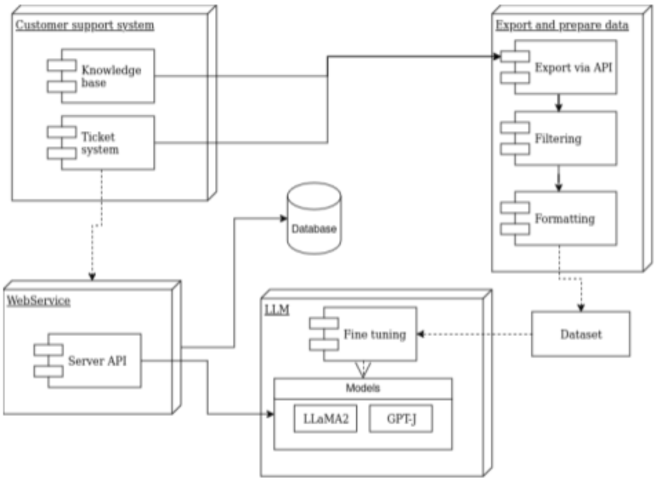
Методологія

Для постнавчання великих мовних моделей використовувався метод Q-LoRA - Quantized Low Rank Adapters. Основними принципами якого є:

- Низькорівнева адаптація (Low Rank Adaptation, LoRA) передбачає додавання до параметрів моделі низькорівневих матриць, які навчаються під час тонкого налаштування. Це дозволяє моделі зберігати основні параметри незмінними і тільки коригувати параметри за допомогою невеликих додаткових матриць.
- Квантування (Quantization) що передбачає перетворення параметрів моделі з високоточного формату (наприклад, 32-бітного) в менш точний (наприклад, 4-бітний). Це значно зменшує обсяг пам'яті, необхідний для зберігання і обробки параметрів моделі.

software engineering

Схема системи



Опис програмного забезпечення, що було використано у дослідженні

- Для експорту даних з бази знань, та системи обробки клієнтських звернень був створен API клієнт з використанням мови Python.
- Постнавчання моделей відбувалось в середовищі Google Colab з T4 GPU 15G RAM
- Налаштовані моделі зберігаються в приватном репозіторії Hugging Face
- Вебсервер реалізован за допмогою Python фреймворка FastAPI
- Для інтеграції в систему підтримки клієнтів ZohoCRM було розроблено та реалізовано віджет з використанням Node.JS та zoho-extension-toolkit
- Для збереження статистичних даних використовували СУБД PostgreSQL

Збір даних

При обробці клієнтських запитів менеджери по роботі з клієнтами оцінювали згенеровані відповіді за наступними критеріями:

- Зміст і відповідність завданню
- Граматика і орфографія
- Структура і зв’язність
- Стиль і читабельність

Кожен з критеріїв оцінювався за п’ятибальною шкалою де “1” - найгірший результат “5” - найкращий

software engineering

Збір даних

Оскільки метрика “зміст і відповідність завданню” має вирішальне значення то для розрахунку загальної оцінки був обран метод вагового усереднення.

Наступні ваги були обрані для розрахунку:

– Зміст і відповідність завданню — 0.45;

– Граматика і орфографія — 0.2;

– Структура і зв’язність — 0.2;

– Стиль і читабельність — 0.15.

software engineering

Результати експерименту

Під час експерименту було зібрано 124 оцінки щодо згенерованих відповідей на різні клієнтські запити, надані постнавченеми моделями LLaMa2, GPT-J та Chat GPT v3.5

Назва метрики	GPT-J-B6	LLaMa-2-7B	Chat GPT v 3.5
Зміст і відповідність	1,335483871	1,569512195	1,42983871
Граматика і орфографія	0,5967741935	0,6790322581	0,6887096774
Структура і зв'язність	0,6258064516	0,7258064516	0,6193548387
Стиль і читабельність	0,4233870968	0,5431451613	0,4221774194
Сумарне значення	2,981451613	3,517496066	3,160080645

Аналіз отриманих результатів

Після аналізу результатів ми можемо зробити висновок, що Chat GPT v 3.5 дає незадовільні результати, де приблизно 40% відповідей не можуть бути застосовані в конкретній галузі. Це вимагає додаткового налаштування з використанням даних, пов'язаними з галуззю, або подальшого навчання на основі підказок.

LLaMA-2-7B перевершує GPT-J-6B та надає релевантні результати, вимагаючи менше обчислювальних ресурсів для додаткового налаштування. Але все одно існує близько 20% пропозицій від LLaMA-2-7B які неможливо застосувати в практичній роботі.

Публікація результатів

Результати роботи представлені під час XXVIII міжнародного молодіжного форуму «РАДІОЕЛЕКТРОНІКА ТА МОЛОДЬ У XXI СТОЛІТТІ»

Підготовлені та подані матеріали для представлення на конференції Computer Science and Information Technologies (CSIT)

Підсумки

Після аналізу результатів ми можемо зробити висновок, що модель, навчена на загальних даних (ChatGPT), дає незадовільні результати, де приблизно 40% відповідей не можуть бути застосовані в конкретній галузі, та постнавчання необхідне.

Використання великих мовних моделей з відносно малою кількістю параметрів дозволяє проводити постнавчання швидко, та з використанням доступних середовищ, але малі моделі генерують менш зв'язний текст, з більшою кількістю граматичних та орфографічних помилок.

Існують інші моделі які можна використовувати для постнавчання і аналізу.