This is a Sample Title

Sangeeta Oswal[a], Aum Kulkarni[b], Ritesh Bhalerao[b], Sayali Kawatkar[b], Dyotak Kachare[b]

[a]*Assistant Professor, AI & Data Science, VESIT, Mumbai*
[b]*Student, AI & Data Science, VESIT, Mumbai*

Abstract

Abstract text.

Keywords:

1. Introduction

Language models, particularly LLMs, are widely used and adopted for various applications in multiple domains. They have shown immense effectiveness in every natural language processing task, and generalized models like GPT-4 [1], Gemini [2], Claude [3], LLaMA 2 [4], and Mistral [5] have shown capabilities unseen by any technology. This impact is also apparent in the ECommerce domain, with LLMs dominating with remarkable performance through specialized models like EComGPT [6] and LiLiuM [7], thereby contributing to enhancing the efficiency of businesses. On average, there are about 40 MSMEs, with a median of 31 per 1,000 people [8]; Adoption of Language Models can help them improve efficiency and productivity, integrate technology and enhance customer enhancements, a view held by MSMEs themselves [9]. However, in most cases, adopting these models is not feasible for these organizations due to the costs of training such huge models, data security, and privacy concerns [10].

Small Language Models (SLMs) have fewer parameters than LLMs, which can range up to hundreds of billions of parameters [11]. In particular, we are exploring models in the range of 300 million to 2 billion parameters [12]. While these models cannot compete with LLMs regarding generalization, they are effective when fine-tuned for specificity [13]. SLMs drastically reduce the cost and hardware resources required to create language models capable of completing such domain-specific tasks [14, 15].

This study provides empirical evidence demonstrating that Small Language Models (SLMs) achieve superior performance with minimal fine-tuning across a diverse range of e-commerce tasks. Consequently, MSMEs can adopt or independently develop language models tailored to their specific requirements. This approach addresses key challenges related to computational overhead, cost efficiency, and data security, enabling broader accessibility and customization.

Through this paper we make the following contributions: (1) We show that for the E-Commerce domain, the metrics for the tasks answered by SLMs were comparable to LLMs, thus making them viable replacement for them without any major impact to performance in tasks, (2) Perform experiments and provide a comparative analysis between some of the SLMs and other LLMs, (3) MSMEs have a viable option to adopt Language Models, drastically reducing the resources required to create such high performing models on their own.

We first discuss existing work in this area, showcasing how SLMs have found success in other domains and how they can be used on customer devices to reduce the dependence on cloud resources. Additionally, we discuss the prevalence of language models in the domain of e-commerce. Section 3.1 discusses the features of the dataset, fine-tuning details, and the algorithms and methods used for reducing the number of parameters. The Results in 4.2 summarize the performance of SLMs compared to other language models.

2. Related Work

Small Language Models (SLMs) are changing the AI landscape, proving that smaller can indeed be faster and smarter. SLMs are more efficient, inexpensive, and flexible compared with their larger counterparts, which usually require huge computational capabilities and face problems related to non-compliance and security. Studies like those of Sinha et al. (2024) highlight that SLMs show significant practical utility by performing within 10% of cutting-edge large models such as GPT-4o-mini, Gemini-1.5-Pro, and DeepSeek-v2 in a variety of tasks, domains, and reasoning types [13]. This makes SLMs a practical solution, particularly in resource-constrained environments due to faster inference and the ability for edge-device deployment. Chen et al. (2024) showcased the effectiveness of domain-specific SLMs by designing OnlySportsLM, a compact 196M parameter model optimized for sports-related tasks. They leveraged specialized datasets and the RWKV-v6 architecture to achieve competitive performance with remarkable efficiency. This approach enabled the model to rival or even exceed the performance of larger general-purpose models, proving that smaller, domain-focused models can deliver competitive results [16]. Pham et al. (2024) introduced SlimLM, which demonstrated the ability to efficiently perform on-device document assistance tasks, showcasing the prospects for SLMs in targeted applications. By determining trade-offs between model size, context length, and inference time, SlimLM was able to efficiently process on a Samsung Galaxy S24. This approach focuses on the aspect that SLMs can greatly reduce dependence on cloud systems, providing affordable and privacy-friendly solutions [14]. Similarly, Sharma et al. (2024) presented ChipNeMo, a model that exceeds the capabilities of larger counterparts like Claude 3 Opus and ChatGPT-4 Turbo while reducing the Total Cost of Ownership (TCO) by 90-95% [10].

Given these advantages, SLMs have the potential to revolutionize a number of sectors, especially eCommerce, where personalization and efficiency are critical. Language models are already automating the generation of product descriptions, solving cold-start problems, and boosting metrics like click-through rates and customer engagement. Herold et al. (2024) showed that in non-English activities, custom LLMs—like eBay's LiLiuM—outperfsorm general-purpose models by providing faster and more accurate text creation [7]. Furthermore, the integration of language models with visual models enhances tasks such as product matching, attribute extraction, and category categorization, leading to improved search and recommendation systems. Y. Li et al. (2023) introduced instruction-tuned models such as EcomGPT, which were trained on specialized datasets designed specifically for eCommerce. These models surpasses larger, general-purpose models like ChatGPT in classification, matching, and text creation due to improved generalization and zero-shot capabilities. In terms of average performance on unseen datasets, EcomGPT, even with the lowest number of parameters (560 million), outperforms ChatGPT, which has over 100 billion parameters [6]. Similarly, Peng Et Al. (2024) advanced the field with open source initiatives like eCeLLM and datasets such as ECInstruct, which improved product matching, attribute extraction, and search categorization [17]. While these developments highlight the immense potential of SLMs, their real-world performance and applicability, particularly for Micro, Small, and Medium Enterprises (MSMEs) in eCommerce, remains underexplored. As compared to larger enterprises, smaller businesses often face barriers such as high costs and a lack of technical expertise. There is a pressing need for further research into how SLMs can be adapted for these businesses, offering them affordable solutions that are easy to integrate with existing systems. By focusing on MSMEs, this study aims to show how smaller models can help businesses improve their operational efficiency and customer experiences, opening the path for affordable and accessible AI solutions.

3. Methodology

3.1. Dataset

ECInstruct is an instruction fine tuning dataset for E-Commerce tasks [17]. The dataset contains a total of 116,528 samples distributed to a toal of 10 tasks. Each of these tasks are aim to test the model on a specific task or area for which LLMs are commonly used for in E-Commerce. All of the 10 tasks can be classified into 4 high level categories based on the type of task that is performed. These tasks along with their description are listed as follows:

3.1.1. Tasks

1. User Understanding:
   (a) Sentiment Analysis (SA)
   Given a product review by a user, identify the sentiment that the user expressed on the product. The task will help models understand what sentiment users express and recommend more proper products to the user.
   (b) Sequential Recommendation (SR)
   Given the interactions of a user over the products, predict the next product that the user would be interested in. By learning on this task, the models will have a comprehensive view of user preferences, which enables models to cater to users' future needs.
2. Product QA
   (a) Answerability Prediction (AP)
   Given a product-related question and reviews of this product, predict if the question is answerable.
   (b) Answer Generation (AG)
   Given a product-related question and reviews as supporting documents, generate the answer to the question.
3. Product Understanding

    (a) Attribute Value Extraction (AVE)
        Given the titles, descriptions, features, and brands of the products, extract values for the specific target attributes. By understanding product attribute values, models can extract key properties of the products and build the profiling for them, which is beneficial in many e-commerce scenarios, such as customer service agents and explanations.
    (b) Product Relation Prediction (PRP)
        Given the titles of two products, predict their relation. Studying the relations between products can help models generate better results when conducting other e-commerce tasks such as recommendations.
    (c) Product Matching (PM)
        Given the titles, descriptions, manufacturers, and prices of the products from two different platforms, predict if they are the same product. This task enables the model to learn the similarities among products.

4. Query Product Matching
    (a) Multiclass Product Classification (MPC)
        Given a query and a product title, predict the relevance between the query and the product (Exact, Substitute, Complement, Irrelevant). This task helps models learn the fine-grained relevance between queries and products, promoting better recommendation results.
    (b) Product Substitute Identification (PSI)
        Given a user query and a potentially relevant product, predict if the product can serve as a substitute for the user's query.
    (c) Query Product Ranking (QPR)
        Given a user query and a list of potentially relevant products to the query, rank the products according to their relevance to the query.

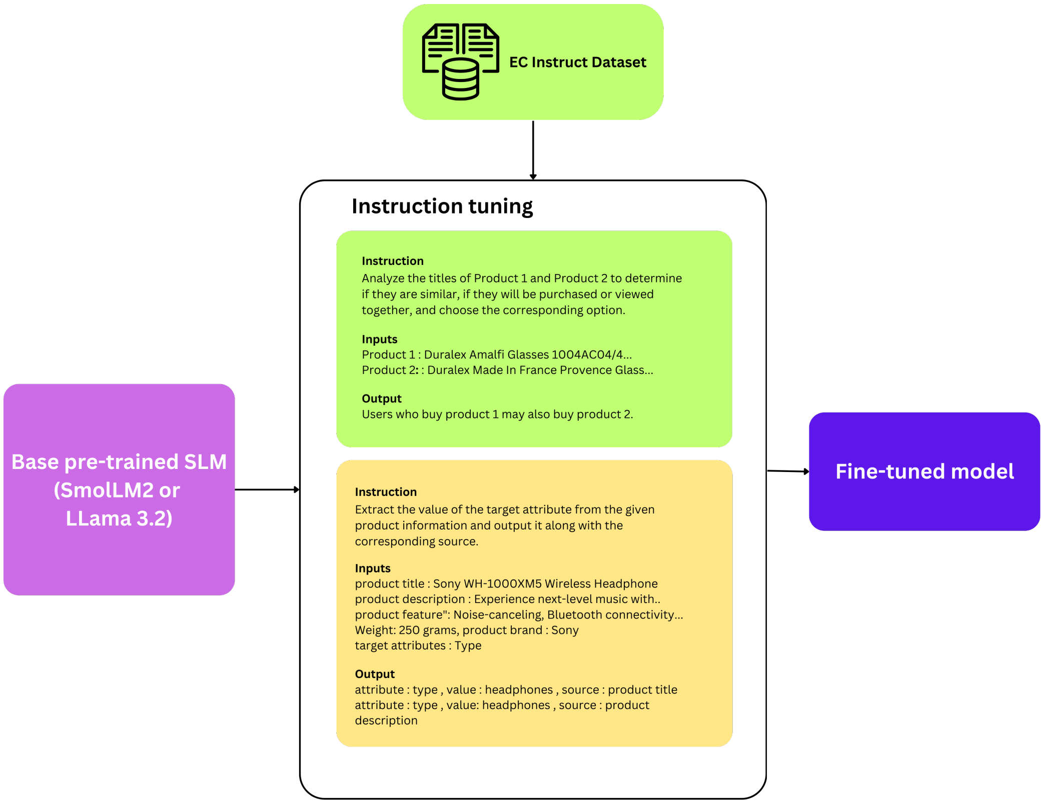
3.1.2. Testing Features

The entire dataset is divided in test, train and validation datasets for each task individually. This is provided as a part of the dataset as a separate column. This allows consistent samples to be trained on every model allowing for a more robust comparison between models instead of randomly splitting them. This also allows for a few more features to the datset to test and gain a realistic measure of model performance. These two features are: **Diverse and Single Instructions Testing** & **In Domain and Out of Domain Testing**

1. Diverse and Single Instructions Testing: The dataset for each of the tasks contain six different types of instruction prompts covering different languages styles, allowing for better generalizability and understanding of tasks. Each task contains 6 diverse instruction prompts, where 1 of these is kept only in the test dataset, thus this prompt isn't seen while training.
2. In Domain and Out of Domain Testing: The In Domain test set contains product and categoried that were present during training the data. The Out of Domain test dataset contains a product category completely unseen by the model allowing to test for the generalizability of the model. This captures the real world applicability of the model as new products and categories emerge when MSMEs expand their business in newer fields.

3.2. Model

This study employs several Small Language Models (SLMs) including SmolLM (v1 and v2) 1.7B and 360M variants, and Llama 3.2 1B [18, 19, 20, 21, 22, 23, 24]. These models are chosen for their performance and efficiency showcased on several benchmarks. They are specifically designed for local hardware and edge devices, balancing cost-effective inference with competitive language understanding. The fine-tuning process is tailored to the ECInstruct dataset, enabling the model to specialize in specific e-commerce-related tasks. Particulary, we instruction tune the pretrained SLMs on the ECInstruct dataset to create high-performing domain specific models, as shown in figure 1.



Instruction
Analyze the titles of Product 1 and Product 2 to determine if they are similar, if they will be purchased or viewed together, and choose the corresponding option.

Inputs
Product 1 : Duralex Amalfi Glasses 1004AC04/4...
Product 2: : Duralex Made In France Provence Glass...

Output
Users who buy product 1 may also buy product 2.

Instruction
Extract the value of the target attribute from the given product information and output it along with the corresponding source.

Inputs
product title : Sony WH-1000XM5 Wireless Headphone
product description : Experience next-level music with..
product feature": Noise-canceling, Bluetooth connectivity...
Weight: 250 grams, product brand : Sony
target attributes : Type

Output
attribute : type , value : headphones , source : product title
attribute : type , value: headphones , source : product description

Figure 1: Instruction tuning

As mentioned in the Introduction, Micro, Small, and Medium Enterprises (MSMEs) lack the budget or resources to avail training on high end GPUs. To emulate such a situation we fine tuned our pre-trained models using QLoRA based on LoRA (Low-Rank Adaptation of Large Language Models), the most widely used PEFT (Parameter Efficient Fine Tuning) technique drastically reducing the computational requirements for training. Parameter Efficient Fine-Tuning (PEFT) techniques have emerged as a crucial strategy for adapting large language models (LLMs) to new tasks without the need for extensive computational resources. These methods focus on updating a minimal set of parameters, thereby reducing both memory and computational demands. PEFT techniques like LoRA and its variants have been instrumental in making fine-tuning more accessible and efficient [25, 26, 27]. Table 1 summarizes the total and trainable parameters of the fine-tuned models. The naming convention for the these models follows the format *base_model_name-EC*, where *base_model_name* represents the underlying base model used as the foundation for fine-tuning.

Table 1: Model Parameters

Model	**Total params**	**Trainable params**
smolLM-EC	1.7B	19M
smolLM-EC	360M	9.4M
smolLM2-EC	1.7B	19M
smolLM2-EC	360M	9.4M
Llama3.2-EC	1B	13M
Llama3.2-EC	3B	26.4M

LoRA is a prominent PEFT method that approximates model changes using low-rank matrices. It freezes the original model weights and updates only the low-rank adapters, which significantly reduces the number of trainable parameters. This approach has been shown to be effective across various tasks and models, providing a balance between performance and resource efficiency [28]. However, LoRA can sometimes underperform compared to full fine-tuning, especially in complex domains, prompting the development of more advanced techniques like HydraLoRA and PiSSA [29, 30]. The figure 2 shows how LoRA is used in transformer blocks. QLoRA works upon this by quantizing the pre trained model weights and adding the adapter weights which are then updated during training [31]. This drastically reduces the memory requirements of training the model making it feasible to complete training of models within the budget and resource constraints of MSMEs. This study made use of the PEFT library's implementation of QLoRA for fine tuning.

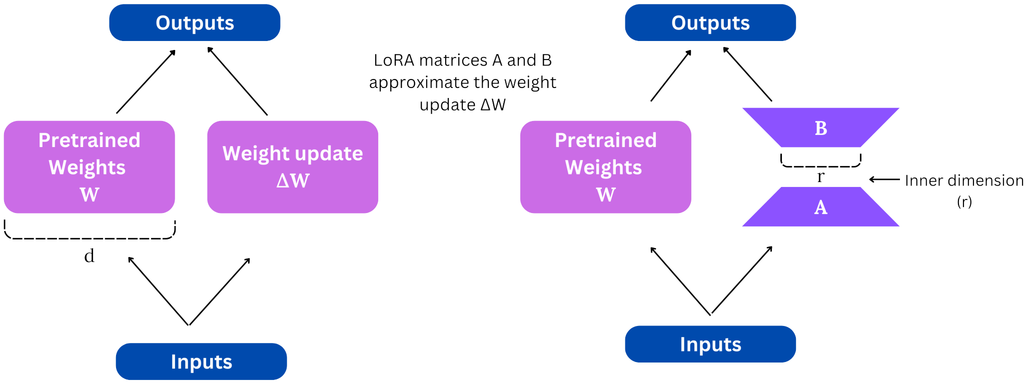


Figure 2: Fine-tuning (left) v/s LoRA Fine-tuning (right)

4. Experimental Outcomes

4.1. Evaluation setup

General-purpose LLMs are evaluated using checkpoints given by their authors. GPT-4 Turbo [1], Gemini Pro [2], and Claude 2.1 [3] are accessed via official APIs, whereas Llama-2 13B-chat [4] and Mistral-7B Instruct-v0.2 [5] are retrieved from Hugging Face. Since in-context examples are known to improve results, the assessment uses a 1-shot setting to balance computational expense and performance. Because of their slowness and tendency to lower user interest, extensive prompt engineering and few-shot learning are considered impracticable for large-scale e-commerce applications. Also fine tuning the model for the specific tasks of the business eliminates the necessity of employing any such prompt engineering techniques.

EcomGPT is evaluated using its checkpoint, which was released by the authors [6]. Since 1-shot empirically demonstrates better performance than 0-shot for EcomGPT, both 0-shot and 1-shot evaluations are carried out. For every assignment, the best performance from these assessments is reported. Evaluation results for both General Purpose LLMs and E-Commerce LLMs are taken as it is from the previous works of authors of ECInstruct[17].

4.2. Results

We evaluated our model on different tasks after fine-tuning it on the entire dataset for all tasks. Early results show promise as the model exhibits comparative or superior performance over much larger models than its size, in terms of parameter count and overall architecture. This comparison is with respect to other models benchmarked on the EC-Instruct dataset [17]. It encompasses several tasks aimed at improving both product and user understanding as explained in section 3.1.1. Following are preliminary results for the in-domain evaluation of some tasks from the EC-Instruct.

Table 2: Overall Performance in IND Evaluation

Model	**AVE**	**PRP**	**PM**	**SA**	**SR**	**MPC**	**PSI**	**QPR**	**AP**	**AG**
	F1*	**Macro F1**	**F1**	**Macro F1**	**HR@1**	**Accuracy**	**F1**	**NDCG**	**F1**	**F_{BERT}**
GPT-4 Turbo	0.495	0.326	0.753	0.516	0.387	0.611	0.195	**0.875**	0.649	0.858
Gemini Pro	0.396	0.136	0.867	0.470	0.269	0.584	0.248	0.821	0.506	0.855
Claude 2.1	0.381	0.275	0.523	0.415	0.066	0.655	0.273	0.821	0.280	0.841
Llama-2 13B-chat	0.002	0.323	0.434	0.188	0.056	0.504	0.252	0.815	0.623	0.811
Mistral-7B Instruct-v0.2	0.369	0.324	0.613	0.470	0.164	0.529	0.305	0.842	0.588	0.853
EcomGPT	0.000	0.091	0.648	0.188	0.042	0.540	0.170	0.000	0.086	0.669
SmolLM2-1.7B-EC	**0.988**	**0.596**	**0.995**	**0.604**	**0.481**	**0.666**	**0.373**	0.869	**0.860**	**-**

4.2.1. Metrics

F_1^* is used for Attribute Value Extraction to evaluate the balance between precision and recall, ensuring an effective measure of the model's performance. The equations for precision*, recall*, and F_1^* are defined in equation 1, where the model can predict null value *(NV)* or incorrect value *(IV)* for negative samples. For positive samples, the model predictions can be correct value *(CV)*, wrong value *(WV)*, and null value *(NL)*. Moreover, normal F_1 score is used to evaluate the performance on product matching (PM), product substitute identification (PSI), and answerability prediction (AP). For sentiment analysis (SA) $Macro\,F1$ is used. Sequential recommendation (SR) is evaluated on hit rate at 1 *(HR@1)*, which is a very popular metric in sequential recommendation that calculates whether the top-ranked product for a certain user is relevant. F_{BERT} evaluates the quality of generated texts by measuring the similarity between the embeddings of tokens in the generated text and the ground-truth text.

$$\text{precision}^* = \frac{NV + CV}{NV + IV + CV + WV} \qquad \text{recall}^* = \frac{NV + CV}{N} \qquad F_1^* = 2 \times \frac{\text{precision}^* \times \text{recall}^*}{\text{precision}^* + \text{recall}^*} \tag{1}$$

$$P_t = \frac{TruePositives}{TruePositives + FalsePositives} \qquad R_t = \frac{TruePositives}{TruePositives + FalseNegatives} \tag{2}$$

$$\text{Macro-F}_1 = \frac{1}{|T|} \sum_{t \in T} \frac{2 P_t R_t}{P_t + R_t} \tag{3}$$

4.3. Discussions

It is evident from the results that much smaller models, with minimal finetuning, outperforms some of the top LLMs in the current market. This indicates potential for lots of possible applications for these SLMs in various domains with modest requirements. This also aligns with a new uprising direction in LLM research, where efficiency is underscored. Using smaller language models for niche tasks can make the whole process economical along with wider ranges of devices supporting the models because of on-edge processing.

References

[1] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al., Gpt-4 technical report, arXiv preprint arXiv:2303.08774 (2023).

[2] G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican, et al., Gemini: a family of highly capable multimodal models, arXiv preprint arXiv:2312.11805 (2023).

[3] Anthropic, Model card for claude 2, `https://www-cdn.anthropic.com/bd2a28d2535bfb0494cc8e2a3bf135d2e7523226/Model-Card-Claude-2.pdf`, accessed: 2025-01-07 (2023).

[4] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al., Llama 2: Open foundation and fine-tuned chat models, arXiv preprint arXiv:2307.09288 (2023).

[5] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. d. l. Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, et al., Mistral 7b, arXiv preprint arXiv:2310.06825 (2023).

[6] Y. Li, S. Ma, X. Wang, S. Huang, C. Jiang, H.-T. Zheng, P. Xie, F. Huang, Y. Jiang, Ecomgpt: Instruction-tuning large language models with chain-of-task tasks for e-commerce, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 38, 2024, pp. 18582–18590.

[7] C. Herold, M. Kozielski, L. Ekimov, P. Petrushkov, P.-Y. Vandenbussche, S. Khadivi, Lilium: ebay's large language models for e-commerce, arXiv preprint arXiv:2406.12023 (2024).

[8] K. Haider, M. Khanna, M. Kotei, K. Kushnir, S. Singh, T. Sridhar, Micro, small and medium enterprises - economic indicators (msme-ei), Tech. rep., International Finance Corporation (2019).

[9] S. Gowda, Ai catalyst: Cracking the code for msme productivity (2024).

[10] A. Sharma, T.-D. Ene, K. Kunal, M. Liu, Z. Hasan, H. Ren, Assessing economic viability: A comparative analysis of total cost of ownership for domain-adapted large language models versus state-of-the-art counterparts in chip design coding assistance (2024). `arXiv:2404.08850`.
URL `https://arxiv.org/abs/2404.08850`

[11] E. B. Loubna Ben Allal, Anton Lozhkov, The rise of small language models (slms), Hugging Face Blog (2023).
URL `https://huggingface.co/blog/smollm`

[12] S. Beatty, The phi-3: Small language models with big potential, Microsoft Source Feature (2023).
URL `https://news.microsoft.com/source/features/ai/the-phi-3-small-language-models-with-big-potential/`

[13] N. Sinha, V. Jain, A. Chadha, Are small language models ready to compete with large language models for practical applications?, arXiv preprint arXiv:2406.11402 (2024).

[14] T. M. Pham, P. T. Nguyen, S. Yoon, V. D. Lai, F. Dernoncourt, T. Bui, Slimlm: An efficient small language model for on-device document assistance, arXiv preprint arXiv:2411.09944 (2024).

[15] R. Yi, X. Li, W. Xie, Z. Lu, C. Wang, A. Zhou, S. Wang, X. Zhang, M. Xu, Phonelm: an efficient and capable small language model family through principled pre-training, arXiv preprint arXiv:2411.05046 (2024).

[16] Z. Chen, C. Li, X. Xie, P. Dube, Onlysportslm: Optimizing sports-domain language models with sota performance under billion parameters, arXiv preprint arXiv:2409.00286 (2024).

[17] L. Peng, et al., ecellm: Generalizing large language models for e-commerce from large-scale, high-quality instruction data, Paper (2024).

[18] L. B. Allal, A. Lozhkov, E. Bakouch, Smollm - blazingly fast and remarkably powerful (Jul 2024).
URL https://huggingface.co/blog/smollm

[19] P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, O. Tafjord, Think you have solved question answering? try arc, the ai2 reasoning challenge, arXiv:1803.05457v1 (2018).

[20] T. Mihaylov, P. Clark, T. Khot, A. Sabharwal, Can a suit of armor conduct electricity? a new dataset for open book question answering, in: EMNLP, 2018.

[21] R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi, Y. Choi, Hellaswag: Can a machine really finish your sentence?, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019.

[22] A. Talmor, J. Herzig, N. Lourie, J. Berant, CommonsenseQA: A question answering challenge targeting commonsense knowledge, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4149–4158. arXiv:1811.00937, doi:10.18653/v1/N19-1421.
URL https://aclanthology.org/N19-1421

[23] D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, J. Steinhardt, Measuring massive multitask language understanding, Proceedings of the International Conference on Learning Representations (ICLR) (2021).

[24] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, C. Hesse, J. Schulman, Training verifiers to solve math word problems, arXiv preprint arXiv:2110.14168 (2021).

[25] A. Chavan, Z. Liu, D. Gupta, E. Xing, Z. Shen, One-for-all: Generalized LoRA for parameter-efficient fine-tuning (2023). arXiv:2306.07967.

[26] M. Thakkar, Q. Fournier, M. D. Riemer, P.-Y. Chen, A. Zouaq, P. Das, S. Chandar, A deep dive into the trade-offs of parameter-efficient preference alignment techniques (2024). `arXiv:2406.04879`.

[27] S. Chen, Y. Ju, H. Dalal, Z. Zhu, A. Khisti, Robust federated finetuning of foundation models via alternating minimization of LoRA (2024). `arXiv:2409.02346`.

[28] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, Lora: Low-rank adaptation of large language models (2021). `arXiv:2106.09685`.
URL `https://arxiv.org/abs/2106.09685`

[29] F. Meng, Z. Wang, M. Zhang, PiSSA: Principal singular values and singular vectors adaptation of large language models (2024). `arXiv:2404.02948`.

[30] C. Tian, Z. Shi, Z. Guo, L. Li, C. Xu, HydraLoRA: An asymmetric LoRA architecture for efficient fine-tuning (2024). `arXiv:2404.19245`.

[31] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer, QLoRA: Efficient finetuning of quantized LLMs (2023). `arXiv:2305.14314`.