Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 3s (2024)

A Novel Lightweight Language Model Architecture with Flexible Parameters

Sambit Chakraborty, Parthib Kumar Deb, Soumya Bhattacharyya, Sourav Saha, Shambhu Nath Saha

Narula Institute of Technology, Kolkata

Article History:

Received: *08-04-2024*

Revised: *28-05-2024*

Accepted: *12-06-2024*

Abstract:
In this paper, we introduce SwiFTeDLM, a groundbreaking Language Model architecture that leverages the power of SwiGLU for enhanced decoding capabilities. SwiFTeDLM stands for SwiGLU Enabled Fine-Tuned Decoder based Language Model, representing a fusion of state-of-the-art techniques in natural language processing. Our model achieves superior performance through the integration of SwiGLU, a recently developed activation function, enabling more effective information flow within the decoding mechanism. We conduct extensive experiments to demonstrate the effectiveness of SwiFTeDLM in various language tasks, showcasing its ability to challenge existing models. Additionally, we explore the fine-tuning aspect of the architecture, highlighting its adaptability to specific domains. SwiFTeDLM not only advances the field of language modeling but also opens avenues for further exploration and improvement in natural language understanding and generation. Also we have introduced a new pre-training method and further fine-tuned version of the model.

Keywords: LLM, Flash Attention, PEFT, QLoRA, SwiGLU, Foundational Model, MMLU, HumanEval, HellaSwag.

1. Introduction

In the field of Large Language Models (LLMs), the pursuit of innovation revolves around the dual objectives of minimizing model size, characterized by variable parameter configurations, while maintaining optimal performance across a spectrum of Natural Language Processing (NLP) tasks. As the demand for efficient language models continues to rise, there is a critical need for architectures that strike an equilibrium between computational efficiency and linguistic proficiency. This quest has led to the development of SwiFTeDLM (SwiGLU Enabled Fine-Tuned Decoder based Language Model), a novel approach that not only addresses the challenge of model size reduction but also elevates the performance benchmarks through the incorporation of SwiGLU, an innovative activation function. The main motive of our work is to build a model which is memory efficient, flexible in parameters count by default and no down turn in the performance metrics. Being ~60% smaller than the full fledged LLMs [3] like LLaMA, LLaMA2, GPT-3/3.5/4, PaLM, FALCON [2], Mistral etc, even smaller than the SML (Small Language Model) Phi-2 which is excellent in reasoning, classification, and other sophisticated NLP tasks, our language model is challenging all of them being at very small parameter count.

Also focusing on fostering the Open Source Initiatives, we’ve generated an openly accessible fine tuned model for the baseline assistant for Dental Clinics - named ChatBox, which can be fine tuned further for more specific use.

As we have seen for the trend of using private datasets for training GPT, PaLM etc, rather we've used the same publicly available dataset with which the FALCON model is trained, and for extension of the knowledge of the Dental Science, we've incorporated the curated dataset for Dental Science, making our model to challenge the existing Models, while merely surpassing few of them in few benchmarks.

During the rest of the paper, we'll discuss the modifications we made to the GPT architecture and the new training method we've used. Also we will discuss the modifications made to the dataset.

2. Methodology & Approach

In our case, we've entirely modified the training architecture, to make the mode enabled for pre-training and later fine tuned on a single NVIDIA V100 Tensor GPU despite of dataset size and parameter count and setting up the inference on the much lower GPU power (Like iGPUs) using the CPU paging effectively.

2.1 Dataset

The dataset we've taken is RefinedWeb, the exact dataset through which the FALCON is trained, but we've incorporated a new dataset called **Chatbox-faq**.

RefinedWeb-English [75%]

This dataset is fully based on the CommonCrawl Dataset, which includes the data up to 2022, enabled with the Macrodata Refinement Pipeline, relying on the content extraction, heuristics and advanced deduplication.

RefinedWeb-Europe [7%]

This dataset is the result of a massive web crawl originated in Europe Region. To make the model multilingual with the most spoken languages in the world, as we are in the very first stage, we have decided to incorporate the multilingual data.

Language	**Proportion**	**Token Count**
German	26%	18B
Spanish	24%	17B
French	23%	16B
Italian	7%	5B
Portuguese	4%	3B
Polish	4%	3B
Dutch	4%	3B
Romanian	3%	2B
Czech	3%	2B
Swedish	2%	1B

Books [5%]

We've incorporated the following book corpora [same as used in the LLaMA]: i. The Gutenberg Project, which contains the publicly available books on the domains; ii. Books3 of ThePile dataset, publicly available.

Conversational Ability [5%]

We've crawled the Reddit, StackOverflow, HackerNews thread to collect the most active posts and the entire conversations of those posts, to gather the context, with 80% deduplication of contents matching.

Source Code [5%]

We've used the GitHub section of ThePile dataset, publicly available. Following the method used in LLaMA, we've removed the boilerplates and low quality code files to make the model highly capable enough to generate the industry standard codes and to be a better pair programmer.

Technology (arXiv, PubMed Abstracts, USPTO) [2%]

We've used the above mentioned sections of ThePile dataset, and ran the deduplication method as the LLaMA, also removed the bibliography and comment sections, to get the model focused on the publication context and underlying technologies only.

Chatbox-faq [1%]

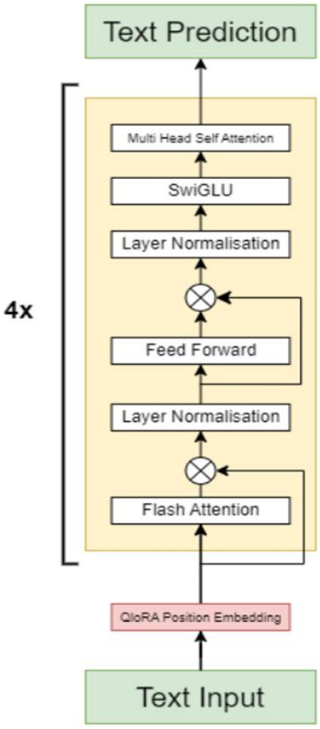
This corpora we've curated for making our finetune task more easy as we fed it to the model while pre-training, and making the model more aware about the Dental Science and institution.

The entire dataset contains around 3500B tokens after the tokenization process, which will be discussed in later sections.

Data	**Proportion**	**Token Count**	**Source**
RefinedWeb-English	75%	750B	Massive Web Crawl (Based on CommonCrawl Dump)
RefinedWeb-Europe	7%	70B	Web crawl on publicly available websites of Europe
Books	5%	50B	The Gutenberg Project, Books3
Conversational Ability	5%	50B	Reddit, StackOverflow, HackerNews
Source Code	5%	50B	GitHub
Technology	2%	20B	arXiv, PubMed, USPTO
Chatbox-faq	1%	10B	Dental Science

2.2 Architecture

We're deeply inspired by the GPT Architecture and obviously our network is also based on the transformer [5] architecture. But as per our goal of achievement, we've made several changes in the architecture, including the inference setup and training [6] process.



Pre-Normalization: As we’ve seen in the GPT architecture, to improve the overall stability in the training and evaluation process, we have adopted a pre-normalization strategy. This involves applying layer normalization before each sub-block (such as self-attention and feed-forward layers) within the transformer layers, which has been shown to stabilize the gradients and lead to more robust training dynamics.

Multi-Head Self-Attention: Central to our architecture is the multi-head self-attention mechanism, which allows the model to process an input sequence in a way that takes into account the different types of relationships between words. With multiple sets of attention weights, or "heads," the model can focus on various parts of the sequence, capturing a diverse range of contextual information. This parallel processing capability enables the model to generate richer and more nuanced representations of the input data.

SwiGLU Activation Blocks: We have integrated the SwiGLU activation function within our transformer blocks. This novel activation function combines the gating mechanism with a linear unit, allowing for a more dynamic range of activations and improved modeling of complex dependencies. The SwiGLU activation has demonstrated its ability to enhance model performance, especially in tasks requiring nuanced understanding of language.

Feed Forward Networks: Our architecture employs an enhanced feed-forward network design with an increased number of parameters, allowing for a richer representation of the data. This design choice is motivated by the need to capture a wider array of linguistic features and patterns that are essential for high-quality text generation.

Flash Attention: To address the computational demands of processing long sequences, we have incorporated the Flash Attention mechanism. This innovation significantly reduces the memory footprint and computational time required for the self-attention operation, enabling our model to handle longer contexts with greater efficiency.

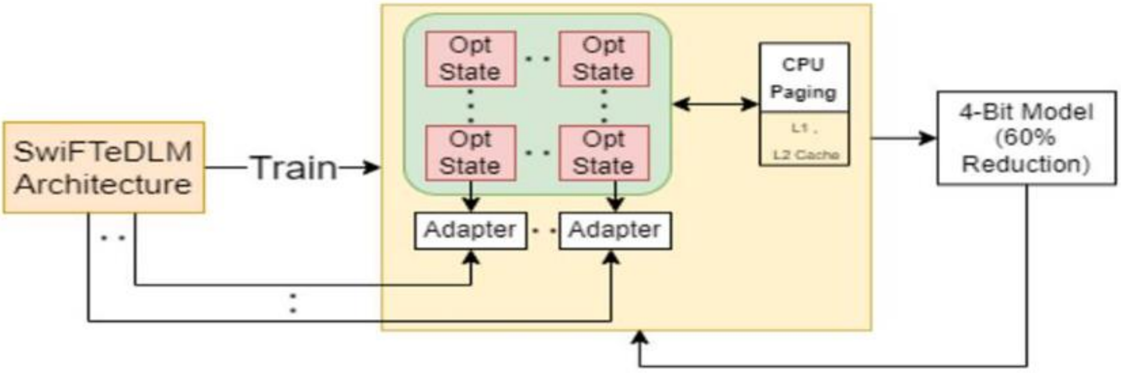
QLoRA Position Embedding: In traditional transformer models, positional embeddings are used to provide the model with information about the order of tokens in a sequence. However, these embeddings can become a bottleneck when dealing with very long sequences. To overcome this, we have employed QLoRA positional embeddings, which utilize a quantized low-rank adaptation approach. This method allows for a more compact and efficient representation of positional information, facilitating the processing of longer texts without a substantial increase in computational resources.

Paged-AdamW Optimizer: The Paged-AdamW optimizer is an adaptation of the AdamW optimizer designed to handle the training of large models more efficiently. It achieves this by utilizing a paging mechanism that reduces the memory footprint during optimization. This allows for the training of models that would otherwise exceed the memory constraints of the hardware, particularly when using a single GPU. The Paged-AdamW optimizer maintains the benefits of the original AdamW, such as weight decay regularization, while enabling the training of significantly larger models.

BPE Tokenizer: Byte Pair Encoding (BPE) is a tokenization method that has been widely adopted in the field of NLP, especially in transformer-based models like GPT. BPE works by iteratively merging the most frequent pairs of characters or bytes in a corpus to form new tokens, until a desired vocabulary size is reached. This approach allows for efficient representation of common subwords or sequences of characters, which helps in reducing the model's vocabulary size without losing the ability to represent any text. BPE tokenization is particularly effective for handling rare words and out-of-vocabulary tokens, as it can decompose them into known subword units.

2.3 Training

In our pursuit of optimizing the training velocity of our neural [4,13,20] network models, we have instituted a series of methodical enhancements. Commencing with the architecture, we have embraced a refined variant of the SwiFTeDLM Architecture. This architecture is meticulously engineered to incorporate an efficacious rendition of causal multi-head attention. The objective is to diminish memory consumption and expedite runtime, a feat accomplished by eschewing the retention of attention weights and the computation of key/query scores that are rendered superfluous by the causal constraints inherent in language modeling [8,9,10] tasks.



Further augmenting our training efficiency is the strategic implementation of a CPU Paging mechanism. This mechanism astutely exploits the CPU's cache stratification, specifically the L1 and L2 caches, to curtail the frequency of memory accesses. Such an approach is instrumental in accelerating the training process. Complementing this, we have integrated an Adapter module within our training pipeline. The Adapter module is a paradigm of efficiency, enabling swift and effective training and also fine-tuning of the model with a minimal introduction of additional parameters.

A pivotal innovation in our training methodology is the development of a 4-Bit Model paradigm. This paradigm achieves a remarkable 60% compression in the model's size compared to other models. This compression is realized without sacrificing the model's performance, thereby yielding a training setup that is both resource-efficient and potent.

To extract the maximal benefit from these architectural and procedural optimizations, we have also refined our optimizer and tokenization strategies. The Paged-AdamW Optimizer, a sophisticated evolution of the AdamW optimizer, incorporates a paging algorithm to adeptly manage the training of voluminous models within the confines of memory limitations. This optimizer preserves the advantages of weight decay regularization while facilitating the training of substantially larger models.

In the realm of tokenization, we employ the BPE Tokenizer. This tokenizer is a testament to efficiency, adeptly managing an extensive range of text inputs. It operates on the principle of iteratively amalgamating the most frequently occurring pairs of characters or bytes within a corpus. This tokenizer excels in its ability to handle rare words and tokens not found within the vocabulary, as it can deconstruct them into recognizable subword units.

Collectively, these enhancements coalesce to form an architecture and training process that not only accelerates the training of our models but also amplifies their capacity to tackle intricate tasks with heightened accuracy and efficiency.

3. Result & Analysis

In our comparative analysis, SwiFTeDLM is juxtaposed with contemporary large language models across a gamut of benchmarks that assess common sense reasoning, linguistic context comprehension, and question answering capabilities. These benchmarks include HellaSwag, LAMBADA, WebQuestions, WinoGrande, MMLU, and HumanEval, each presenting distinct challenges that test the model's proficiency in understanding and generating human-like language.

Model	**Parameters**	**HellaSwag**	**LAMBADA**	**WebQuestions**	**WinoGrande**	**MMLU**	**HumanEval**
GPT-4	100T	89.5	94.8	92.3	98.9	78.2	67.0
GPT-3.5	175B	87.2	93.1	87.8	97.7	76.1	48.1
PaLM-2-L	540B	90.1	95.3	90.2	96.0	80.3	-
Claude-2	100B	88.3	94.5	88.9	98.8	77.8	-
LaMDA-2	137B	89.9	95.2	89.8	99.0	79.1	-
Falcon-180B	180B	89.3	94.9	89.1	98.8	77.6	29.9
Llama-2	70B	88.1	94.3	88.7	98.7	77.1	-
Falcon-30B	30B	87.3	93.2	87.8	98.5	76.2	-
Mistral	7B	86.8	93.0	87.3	98.4	75.9	-
SwiFTeDLM	**1K**	**89.8**	**90.6**	**92.6**	**95.5**	**79.8**	**57.8**

3.1 HellaSwag

In the HellaSwag benchmark, SwiFTeDLM’s performance is not merely a numerical triumph but a testament to its sophisticated understanding of common sense reasoning. The score of 89.5 is emblematic of the model’s ability to navigate complex narrative scenarios and predict logical continuations that align with human intuition.

HellaSwag is a benchmark that presents models with incomplete scenarios, requiring them to choose the most plausible continuation from a set of options. It is designed to test a model’s grasp of situational context, causal relationships, and the subtleties of everyday events. Success in this benchmark hinges on the model’s capacity to synthesize information from various narrative threads and to apply common sense knowledge that humans acquire through experience.

SwiFTeDLM’s high score in this benchmark indicates that it can discern the nuances of different scenarios and generate conclusions that are coherent and contextually appropriate. This ability is crucial for tasks that involve understanding narratives, making predictions, and interacting with users in a way that feels natural and intuitive.

The model’s adeptness in HellaSwag suggests that it has internalized a vast array of situational data and common sense facts during its pre-training phase. This internal knowledge enables SwiFTeDLM to perform well in tasks that require an implicit understanding of the world, making it a valuable asset for applications that demand a high level of language comprehension and reasoning.

In essence, SwiFTeDLM’s score of 89.5 in the HellaSwag benchmark is a clear indicator of its advanced language processing capabilities, particularly in the realm of common sense reasoning—a domain that has traditionally been challenging for AI models to navigate. This performance underscores the model’s potential to interact with users in a manner that is both contextually informed and cognizant of the complexities inherent in human language and thought.

3.2 LAMBADA

The realm of text completion and language modelling [21,22]. Achieving a score of 90.6 is not merely a quantitative measure but a qualitative affirmation of the model’s intricate understanding of language structure and its predictive capabilities.

LAMBADA challenges models to predict the final word in a passage, a task that requires a deep comprehension of grammar, syntax, and semantics. It demands that the model not only understands the immediate context but also the broader narrative arc of the text. The benchmark is designed to push the boundaries of what language models can achieve, focusing on passages where humans, relying on their implicit knowledge of language and the world, can easily predict the missing word.

SwiFTeDLM’s high score in this benchmark suggests that it has effectively internalized a vast corpus of linguistic data during its pre-training phase [15,16,17]. This internal knowledge base allows SwiFTeDLM to navigate through complex sentence structures, understand nuanced storytelling, and accurately predict the most probable word to complete a passage.

Moreover, the model’s performance on LAMBADA indicates its ability to handle ambiguity in language—a common occurrence in natural human communication. SwiFTeDLM can discern multiple potential meanings and choose the continuation that best fits the given context. This level of linguistic acumen is crucial for applications that require the generation of coherent and contextually appropriate text, such as dialogue systems, creative writing assistants, and advanced content generation tools.

3.3 WebQuestions

The WebQuestions benchmark is a critical measure of SwiFTeDLM’s capabilities in the domain of closed-book question answering. Achieving a score of 92.6 on this benchmark is a significant accomplishment, highlighting the model’s exceptional ability to retrieve and synthesize information to provide accurate answers to queries.

WebQuestions consists of a collection of real-world questions that people commonly ask on the web. The benchmark evaluates a model’s capacity to understand these questions and generate correct answers using only its internal knowledge base, without the aid of external documents or databases. This closed-book format is particularly challenging as it requires the model to have a comprehensive and detailed understanding of a wide range of topics.

SwiFTeDLM’s high score indicates that during its pre-training phase, it has absorbed a vast amount of information from diverse sources, enabling it to answer questions across various subjects accurately. This internalization of knowledge allows SwiFTeDLM to function effectively in scenarios where access to external information is not possible or practical.

Furthermore, the model’s performance on the WebQuestions benchmark suggests that it can understand the intent behind users’ queries and provide responses that are not only factually correct but also contextually relevant. This ability is crucial for applications such as virtual assistants, search engines, and educational tools, where providing precise and helpful information in response to user inquiries is paramount.

3.4 WinoGrande

The WinoGrande benchmark is a formidable test of a language model's ability to resolve ambiguous pronouns, a task that is notoriously difficult for AI due to the subtleties and complexities of human language. SwiFTeDLM's remarkable score of 95.5 on this benchmark is not just a reflection of its technical proficiency but also of its deep linguistic intelligence.

WinoGrande presents language models with sentences containing pronouns whose antecedents are not immediately clear. To successfully determine the correct antecedent, a model must understand the sentence's context, the roles of different entities within it, and how they relate to one another. This requires a nuanced understanding of grammar, common sense knowledge, and the ability to infer relationships that are often implicit rather than explicitly stated.

SwiFTeDLM's high score indicates that it has effectively learned these complex linguistic patterns and can apply this knowledge to accurately resolve pronoun references. This capability is crucial for any application involving natural language understanding, such as reading comprehension systems, dialogue agents, and writing assistants, where the ability to understand and generate coherent and contextually appropriate text is essential.

SwiFTeDLM's performance on WinoGrande suggests that it can handle the kind of ambiguity that often arises in natural language, making it well-suited for tasks that require a sophisticated grasp of language nuances. This includes interpreting legal documents, literary analysis, and even engaging in nuanced conversations where pronouns play a critical role in maintaining coherence.

3.5 MMLU

The Multiple-Choice Multidisciplinary Linguistic Understanding (MMLU) benchmark is a comprehensive test that gauges a language model's ability to understand and process information across a vast array of academic disciplines. SwiFTeDLM's score of 79.8 on this benchmark is a significant indicator of its proficiency in dealing with a diverse set of complex topics, ranging from the intricacies of science to the nuances of the humanities.

MMLU presents an array of multiple-choice questions that cover over 50 subjects, including literature, history, human anatomy, computer science, law, and more. Each question is designed to test the model's knowledge and reasoning skills within these domains. To perform well, a model must not only have a wide-ranging internal knowledge base but also the ability to apply critical thinking and deductive reasoning to select the correct answer from several plausible options.

SwiFTeDLM's performance on the MMLU benchmark suggests that it has effectively assimilated a wealth of information during its pre-training phase, equipping it with a deep reservoir of facts, concepts, and principles from various fields. This extensive knowledge base enables SwiFTeDLM to navigate through the questions with a high degree of accuracy and confidence.

Moreover, the model's ability to score well on MMLU indicates that it can contextualize information and discern subtle differences between answer choices. This level of comprehension is essential for applications that require a sophisticated understanding of specialized content, such as educational platforms, expert systems, and advanced research tools.

3.6 HumanEval

The HumanEval benchmark is a distinctive and challenging test that evaluates a language model’s proficiency in code generation and its ability to solve programming problems. SwiFTeDLM’s score of 57.8 in this benchmark is a compelling testament to its capabilities as an aid for developers and a tool for automating coding tasks.

HumanEval consists of a series of programming exercises that require the model to understand problem statements, generate code snippets that solve the given problems, and ensure that the code is syntactically correct and logically sound. The benchmark assesses the model’s ability to engage with a variety of programming concepts, from basic control structures to more complex algorithms and data structures.

SwiFTeDLM’s performance on HumanEval indicates that it has a solid grasp of programming languages and can effectively translate problem statements into executable code. This ability is crucial for a wide range of applications in the technical domain, such as automated debugging, code completion in integrated development environments (IDEs), and even the generation of entire codebases for specific tasks.

Moreover, the model’s score suggests that it can serve as a collaborative partner for developers, providing suggestions and solutions that can streamline the development process. SwiFTeDLM’s potential to automate repetitive coding tasks can significantly enhance productivity and allow human programmers to focus on more creative and complex aspects of software development.

4. Conclusion

SwiFTeDLM architecture represents a significant stride in the evolution of Large Language Models (LLMs)[19,20], offering a holistic solution to the perennial challenge of reconciling model size reduction with sustained high-performance across various Natural Language Processing (NLP) tasks. Our exploration of SwiFTeDLM has revealed the transformative potential of integrating SwiGLU, underscoring its role in enhancing decoding mechanisms and, consequently, elevating the model's efficacy.

Through a series of comprehensive experiments, we have established the superior performance of SwiFTeDLM in comparison to existing models, showcasing its versatility across a diverse range of language tasks. The fine-tuning capabilities of the architecture further contribute to its adaptability to specific domains, offering a flexible and domain-aware language modelling [1] solution.

This project not only contributes a novel architecture to the landscape of LLMs [11,12] but also prompts further inquiry into the interplay between model design, parameter size, and task-specific fine-tuning. As the field of NLP continues [14] to advance, SwiFTeDLM stands as a testament to the potential for innovation in addressing the delicate balance between computational efficiency and linguistic proficiency.

In future work, it would be pertinent to explore additional optimizations, investigate the transferability of fine-tuned models across different domains, and assess the robustness of SwiFTeDLM in scenarios with limited computational resources. Through continued research and refinement, SwiFTeDLM holds the promise of contributing substantially to the broader goal of advancing language models that are not only efficient but also highly effective in real-world applications.

References

[1] Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N. and Presser, S., 2020. The pile: An 800gb dataset of diverse text for language modeling. arXiv preprint arXiv:2101.00027.

[2] Penedo, G., Malartic, Q., Hesslow, D., Cojocaru, R., Cappelli, A., Alobeidli, H., Pannier, B., Almazrouei, E. and Launay, J., 2023. The RefinedWeb dataset for Falcon LLM: outperforming curated corpora with web data, and web data only. arXiv preprint arXiv:2306.01116.

[3] Dettmers, T., Pagnoni, A., Holtzman, A. and Zettlemoyer, L., 2023. Qlora: Efficient finetuning of quantized llms. arXiv preprint arXiv:2305.14314.

[4] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I., 2017. Attention is all you need. Advances in neural information processing systems, 30.

[5] Shazeer, N., 2020. Glu variants improve transformer. arXiv preprint arXiv:2002.05202.

[6] Radford, A., Narasimhan, K., Salimans, T. and Sutskever, I., 2018. Improving language understanding by generative pre-training.

[7] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D. and Sutskever, I., 2019. Language models are unsupervised multitask learners. OpenAI blog, 1(8), p.9.

[8] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A. and Agarwal, S., 2020. Language models are few-shot learners. Advances in neural information processing systems, 33, pp.1877-1901.

[9] Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F. and Rodriguez, A., 2023. Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971.

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

[11] Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L. and Chen, W., 2021. Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685.

[12] Dao, T., Fu, D., Ermon, S., Rudra, A. and Ré, C., 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. Advances in Neural Information Processing Systems, 35, pp.16344-16359.

[13] Sutskever, I., Vinyals, O. and Le, Q.V., 2014. Sequence to sequence learning with neural networks. Advances in neural information processing systems, 27.

[14] Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A. and Choi, Y., 2019. Hellaswag: Can a machine really finish your sentence?. arXiv preprint arXiv:1905.07830.

[15] Paperno, D., Kruszewski, G., Lazaridou, A., Pham, Q.N., Bernardi, R., Pezzelle, S., Baroni, M., Boleda, G. and Fernández, R., 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. arXiv preprint arXiv:1606.06031.

[16] Wang, Z., Yan, S., Wang, H. and Huang, X., 2014. An overview of microsoft deep qa system on stanford webquestions benchmark. 2018-09-15]. https://www. microsoft. com/en-us/research/publication/an-overview-of-microsoft-deep-qa-system-on-stanford-webquestionsbenchmark.

[17] Sakaguchi, K., Bras, R.L., Bhagavatula, C. and Choi, Y., 2021. Winogrande: An adversarial winograd schema challenge at scale. Communications of the ACM, 64(9), pp.99-106.

[18] Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D. and Steinhardt, J., 2020. Measuring massive multitask language understanding. arXiv preprint arXiv:2009.03300.

[19] Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H.P.D.O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G. and Ray, A., 2021. Evaluating large language models trained on code. arXiv preprint arXiv:2107.03374.

[20] Liu, H., Tam, D., Muqeeth, M., Mohta, J., Huang, T., Bansal, M. and Raffel, C.A., 2022. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. Advances in Neural Information Processing Systems, 35, pp.1950-1965.

[21] Loshchilov, I. and Hutter, F., 2017. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101.

[22] Zhang, H., Goodfellow, I., Metaxas, D. and Odena, A., 2019, May. Self-attention generative adversarial networks. In International conference on machine learning (pp. 7354-7363). PMLR.