electronics

MDPI

Article

Efficient Inference Offloading for Mixture-of-Experts Large Language Models in Internet of Medical Things

Xiaoming Yuan [1,2,*,†], Weixuan Kong [1,†], Zhenyu Luo [1] and Minrui Xu [3]

[1] Hebei Key Laboratory of Marine Perception Network and Data Processing, Northeastern University at Qinhuangdao, Qinhuangdao 066004, China; 202112001@stu.neuq.edu.cn (W.K.); 2272213@stu.neu.edu.cn (Z.L.)
[2] State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China
[3] School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798, Singapore; minrui001@e.ntu.edu.sg
* Correspondence: yuanxiaoming@neuq.edu.cn
† These authors contributed equally to this work.

Abstract: Despite recent significant advancements in large language models (LLMs) for medical services, the deployment difficulties of LLMs in e-healthcare hinder complex medical applications in the Internet of Medical Things (IoMT). People are increasingly concerned about e-healthcare risks and privacy protection. Existing LLMs face difficulties in providing accurate medical questions and answers (Q&As) and meeting the deployment resource demands in the IoMT. To address these challenges, we propose MedMixtral 8x7B, a new medical LLM based on the mixture-of-experts (MoE) architecture with an offloading strategy, enabling deployment on the IoMT, improving the privacy protection for users. Additionally, we find that the significant factors affecting latency include the method of device interconnection, the location of offloading servers, and the speed of the disk.

Keywords: large language models; efficient inference offloading; mixture-of-experts; Internet of Medical Things

check for updates

Citation: Yuan, X.; Kong, W.; Luo, Z.; Xu, M. Efficient Inference Offloading for Mixture-of-Experts Large Language Models in Internet of Medical Things. *Electronics* **2024**, *13*, 2077. https://doi.org/10.3390/electronics13112077

Academic Editor: Djuradj Budimir

Received: 1 May 2024
Revised: 20 May 2024
Accepted: 24 May 2024
Published: 27 May 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In the medical field, the application of conversational large language models (LLM) has garnered widespread attention to meet the growing demand for personalized healthcare. However, LLMs deployed on the server side still face significant security challenges, such as data breaches and unauthorized access, within the network. Developing an IoMT device that can be deployed on consumer devices is crucial to safeguard user privacy.

The transformer architecture [1] has emerged as a foundational framework for most LLMs, due to its effectiveness for scalability and ability to outperform previous popular neural networks in accuracy. LLMs have been widely applied in various domains. By leveraging edge intelligence, the Internet of Medical Things (IoMT) [2] can collect and analyze medical information, enabling advice delivery through smartphones, wearable devices, and smart home sensors. Deployed in IoMT devices, LLMs can provide users with healthcare services, such as responding to common medical questions and offering health advice. Moreover, it can maximize the protection of user privacy and avoid privacy leaks in the network.

However, they present several significant challenges and limitations in healthcare applications. Given the complex nature of medical data and the specialized knowledge required in healthcare, direct applications of LLMs can pose risks such as misinterpretations and inaccuracies. In particular, general LLMs may struggle to grasp medical terminology and accurately interpret the specific context of medical texts, potentially leading to biased or incorrect medical advice. The data used in LLMs is sourced from different internet platforms, varying significantly in quality. While this approach allows LLMs to gather

vast amounts of information quickly, it also impacts their performance in specific domains, including healthcare. To harness the full potential of LLMs in the medical domain, retraining existing general-purpose LLMs based on high-quality medical datasets is a viable approach. This enables LLMs to attain more accurate language understanding and better generation quality within specific domains. Furthermore, the selection of parameter scale has a profound impact on the performance of LLMs based on the transformer architecture, as a surplus of parameters can cause overfitting, while a shortage can restrict the LLM's performance in medical applications. The resource requirements of LLMs present a challenge in deploying medical LLMs on IoMT devices. LLMs commonly employ a strategy of distributing multiple models across different devices to enhance performance. This distributed approach necessitates communication between multiple devices to coordinate processing and enhance overall system efficiency. In this strategy, different levels of latency can significantly impact the inference speed of distributed LLMs.

To meet individuals' personalized healthcare needs, we aim to establish a new medical LLM specifically designed for deployment on IoMT devices. The literature includes several applications of LLMs in network communications. For example, Xu et al. [3] discuss their application in space–air–ground integrated networks, while [4] explore their use in 6G networks. To integrate theoretical results from these articles and achieve this goal, we use the latest mixture-of-experts (MoE) model from Mixtral, named Mixtral 8x7B [5], for fine-tuning in the medical domain, thus creating a new medical LLM, named MedMixtral 8x7B. Its MoE architecture can enhance performance, leading to higher accuracy in medical questions and answers (Q&As). To deploy MedMixtral 8x7B on IoMT devices, we utilize the methods proposed by [6], incorporating Accelerate's offloading techniques proposed by [7], which allows Mixtral 8x7B to require fewer VRAM and facilitate its deployment on IoMT devices. This ensures better protection of users' privacy data. We conducted extensive evaluations on MedMixtral 8x7B, demonstrating its advantages over strong general large models such as ChatGPT and Llama3 in various aspects of medical Q&As. The efficient offloading architecture saves memory, and research on inference latency suggests key strategies for reducing latency include enhancing disk speed, storing more model weights in the CPU's RAM rather than on disk, and opting for wireless communication.

The contributions of this paper are summarized as follows:

- To obtain efficient medical LLMs for healthcare applications, we fine-tune an LLM based on the MoE architecture, named MedMixtral 8x7B, using medical datasets to meet individuals' personalized healthcare needs.
- To deploy MedMixtral 8x7B on IoMT devices, we propose a novel offloading strategy, which allows the deployment of MedMixtral 8x7B in the IoMT with less resource requirement, thus enhancing the privacy protection for users.
- To assess latency's impact on LLM inference speed, we analyze both local and interconnection communication models. We highlight the critical role of latency in inference processes and propose several strategies to reduce it. These include enhancing disk speed, storing more model weights in the CPU's RAM rather than on disk, and opting for wireless communications.

2. Related Work

2.1. Large Language Models

LLMs, such as the ChatGPT series, have developed significantly in recent years, especially in model architecture, parameter scale, reinforcement learning, and so on. In 2018, OpenAI, in [8], proposes the methods of generative pre-training of a language model. The generative pre-trained transformer (GPT) is a unidirectional autoregressive model. Based on the GPT, OpenAI developed ChatGPT, which is one of the most influential LLM series. To address the issue of requiring fine-tuning layers in GPT-1, GPT-2 underwent training on a larger scale, possessing more parameters, and eliminating the need for fine-tuning layers. In 2019, OpenAI, in [9], proposes GPT-2, using this model, without fine-tuning for specific tasks, it enhances performance across tasks in a logarithm linear manner, and it can

still achieve good results. To enhance contextual understanding and reduce computational overhead, OpenAI released GPT-3. In 2020, OpenAI, in [10], proposes GPT-3, which is trained by 1750B parameters, and achieves high performance on many NLP datasets. In 2022, OpenAI, in [11], designed the reward model (RM), and used reinforcement learning with proximal policy optimization (PPO) to update the GPT model. In 2023, OpenAI proposes GPT-4 [12], which is one of the highest-performing LLMs.

Although the ChatGPT series LLMs have high performance, the latest ChatGPT is not an open LLM. In addition to ChatGPT, many other companies and institutions have also proposed high-performance open LLMs. Zhao et al., in [13], survey the field of LLM, showing the development of LLMs. After the proposal of the transformers architecture, to enhance contextual understanding capability, Devlin et al., in [14], propose a new model, BERT, which is a simple and powerful model that has great results in natural language processing (NLP) tasks. To further improve the performance of BERT, Liu et al., in [15], propose RoBERTa, RoBERTa based on BERT, and introduce several improvements. RoBERTa employs dynamic masking, training with complete sentences without next sentence prediction (NSP) loss, large batch sizes, and larger byte-level byte pair encoding (BPE) for training.

Apart from extending BERT, there are also proposals for different architectures based on transformer. Google proposes the Text-to-Text Transfer Transformer (T5) model in [16]. T5-model modeling every text-related question as a "text-to-text" problem, using the same model, objective function, training, and decoding process for each NLP task, can lead to better performance.

In addition to different model architectures, there are also many models trained based on the traditional transformer architecture. Meta AI, in [17,18], proposes Llama, an open and efficient LLM. Llama is open and free for anyone, allowing individuals to learn about the structure of LLMs from Llama. Additionally, users can fine-tune the model based on Llama. Meta AI, in [19], proposes OPT, which is an open pre-trained transformer LLM. To satisfy the model requirements for the Chinese language, Zhang et al., in [20,21], propose CPM series LLMs, which are Chinese pre-trained LLMs. To meet the needs of generating computer programs, Nijkamp et al., in [22], propose CodeGen, which is a series of LLMs trained on natural language and programming language data, with 16.1B parameters. CodeGen series LLMs have advanced in generating computer programs through input–output examples or natural language descriptions. To fill the gap in open-source models with a large number of parameters, Le et al., in [23], propose BLOOM, which is an open LLM with 1760B parameters. To further expand the usability of multilingual language models, Zeng et al., in [24], propose GLM-130B, which is an English and Chinese pre-trained LLM with 130B parameters. To address the absence of open LLMs based on the MoE structure, Jiang et al. propose Mixtral 8x7B in [5], a model using the MoE architecture, where each token can use 47B parameters but only 13B are used for model inference.

Based on these LLMs, various domain-specific large models have emerged. Due to their conversational capabilities, LLMs are particularly well suited for applications in medical Q&As. Many medical LLMs have demonstrated remarkable abilities in the healthcare field. Luo et al., in [25], propose BioGPT. It is a medical LLM based on GPT-2. BioGPT is first pre-trained on a large-scale biomedical literature dataset, followed by fine-tuning and the adoption of new prompting strategies. It has particularly demonstrated good performance on BC5CDR, KD-DTI, and DDI end-to-end relation extraction tasks. Singhal et al., in [26], present MultiMedQA, providing benchmark datasets for performance evaluation of medical LLMs. Meanwhile, they first propose Flan-PaLM. Based on this, it is fine-tuned into the final product, Med-PaLM, through prompting strategies and other means. In the evaluation, Med-PaLM achieves accuracy comparable to that of real human doctors, yielding excellent results. Building upon this success, they further propose Med-PaLM2 [27]. Med-PaLM2 utilizes an improved base model and employs fine-tuning and prompting strategies specific to the medical domain, improving answer accuracy.

When it comes to medical Q&A tasks, although various types of LLMs are available, none are suitable for deployment on IoMT devices for privacy protection while being capa-

ble of handling such tasks. Typically, models with sufficient performance pose challenges when deployed on IoMT devices, while deployment on the server side introduces network security and privacy protection concerns. LLMs deployable on IoMT devices lack the necessary performance for medical Q&A tasks. To meet the growing demand for medical Q&As and privacy protection requirements from users, designing a high-performance medical LLM deployable on IoMT devices with privacy protection is crucial.

2.2. LLM Efficient Inference Offloading Methods

With the development of LLMs, the size of LLMs is increasing, leading to higher resource requirements for inference. Consequently, more and more scientists are researching ways to reduce resource demands and improve inference speed during LLM inference. Rasley et al., in [28], propose DeepSpeed, which is an optimization library for LLMs. It provides techniques for distributed training, especially memory optimization. Shoeybi et al., in [29], propose Megatron-LM, which is a library for deep learning, and provides a lot of optimization techniques for GPUs. Kwon et al., in [30], propose vLLM, which is a library that provides PagedAttention methods, assisting LLMs with efficient inference. Zhao et al., in [31], propose FSDP; while maintaining the simplicity of data parallelism, it breaks the barrier of slicing the model across multiple processes.

However, these methods do not significantly reduce the occupied VRAM in LLM inference to enable deployment on IoMT devices. Our work greatly improves reducing the occupied VRAM in LLM inference and LLM can be deployed in IoMT devices more easily.

2.3. AI Used in Communication

In the field of communication, the application of AI is rapidly increasing. Han et al., in [32,33], incorporate federated learning into communications yielding promising results. To enhance the management of base station power consumption, Piovesan et al., in [34], designed a machine learning algorithm to assess the power consumption of 5G base stations (BSs).To facilitate the transmission of AI/ML models, Ayed et al., in [35], designed the framework Accordion, which efficiently facilitates the transfer of AI/ML models. In addition to traditional AI, with the development of LLMs there is an increasing number of people applying LLMs in the field of communications. Du et al., in [36], explore using an LLM to assist FPGA wireless signal processing hardware development. Bariah et al., in [37], introduce the application of LLMs in future wireless networks and propose relevant theories along with insights into the challenges LLMs face in communication. Bariah et al., in [38], fine-tune several LLMs for the telecommunications domain language and use them for identifying the 3rd Generation Partnership Project (3GPP) standard working groups. Soman et al., in [39], analyze the capabilities and limitations of integrating LLMs into dialogue interfaces in the telecommunications domain. LLMs demonstrate their effectiveness in various communication applications. However, due to the significant resources required for deployment, there is little discussion about the communication issues of LLMs on IoMT devices. We integrate our offloading model to examine the communication of LLMs among consumer devices.

In mobile networks, Zhang et al. have investigated several security issues. Zhang et al. propose a collaborative mutation-based MTD (CM-MTD) [40] to address the challenges of poor coordination, high network resource consumption, and lack of consideration for future information in MTD. Zhang et al. propose a smart-driven host address mutation (ID-HAM) scheme [41], to address the issues of HAM lacking adaptive adversarial strategies, network states being time-varying, and the oversight of the survivability of existing connections. Zhang et al. propose an intelligent MTD scheme to defend against distributed denial-of-service in SD-IoV [42], to tackle the issues of MTD's inability to handle high-speed dynamic environments, lack of intelligence, and difficulty in tracking.

3. Methods

This section details the design of the communication model, the development of the MedMixtral 8x7B medical LLM, and the implementation of an efficient offloading strategy.

3.1. Communication Model Design

In this section, we investigate the inference offloading latency across IoMT devices, including interconnection communication latency and local communication latency. We model the two types of models separately and calculate the total communication latency at the end.

In e-healthcare, we propose a system model for the IoMT that enables users to utilize medical LLMs. As depicted in Figure 1, each model operates independently on one IoMT device in this system, while possessing the capability to function cooperatively with other models to enhance performance. For interconnection among IoMT devices, there is a set $\mathcal{N} = \{1, \ldots, n, \ldots, N\}$ of IoMT devices, which allows doctors and users to access e-healthcare services via wireless communication. In the interconnection communication model with $\mathcal{N}$, the interconnection communication latency between device n and device $n+1$ is denoted as l_n. The interconnection communication latency between device i and device j is denoted as l_{ij}. The interconnection communication latency is denoted as l_n^{com}; the l_n^{com} can be calculated as

$$l_n^{com} = d_n^p + d_n^{pr} + d_n^q + d_n^w, \tag{1}$$

where propagation delay is denoted as d_n^p, processing delay is denoted as d_n^{pr}, queueing delay is denoted as d_n^q, and waiting delay is denoted as d_n^w.

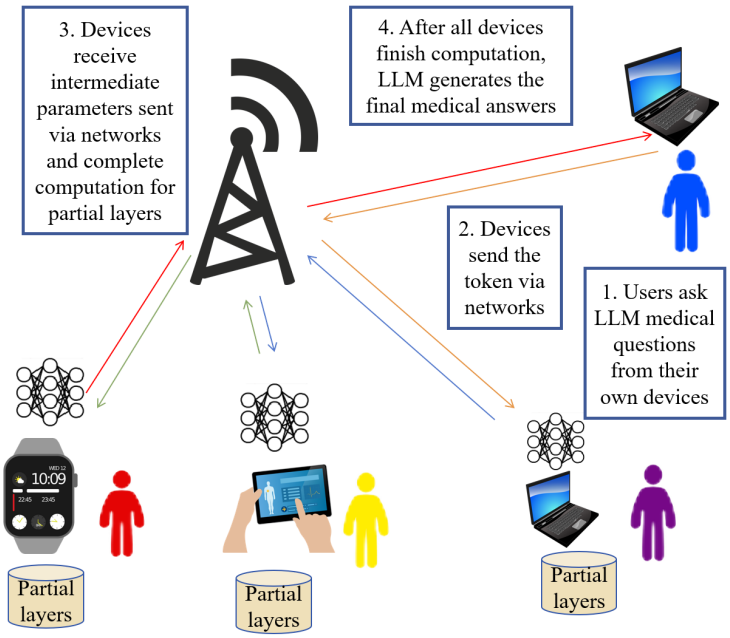

Figure 1. This is the communication model design. Users ask LLM medical questions from their own devices, then devices send the token via networks. Devices receive intermediate parameters sent via networks and complete computation for partial layers, and finally, all devices finish computation; LLM generates the final medical answers to the user.

3.1.1. Interconnection Communication Model

According to Equation (1), the values of d_n^{pr}, d_n^w, and d_n^q can be considered negligible compared to d_n^p. Therefore, the l_n^{com} can be approximately regarded as d_n^p. To obtain the d_n^p, we first need to obtain the signal-to-noise ratio (SNR). In wireless networks, the

signal power between devices is denoted as S_n, and the noise power can be approximately regarded as white Gaussian noise, denoted as ω_n, and the SNR_n is calculated as

$$SNR_n = \frac{S_n}{\omega_n}, \tag{2}$$

where $\omega_n = kTB_n$ is white Gaussian noise, k is the Boltzmann constant, T is the temperature in kelvin, and B_n is the channel bandwidth.

After obtaining the SNR_n, we still need to calculate the channel capacity for obtaining the d_n^p. According to Equation (2), the channel capacity C_n can be calculated as

$$C_n = B_n \log_2(1 + SNR_n), \tag{3}$$

then, according to Equation (3), we calculate the d_n^p. The data volume of medical services is denoted as D_n. The d_n^p can be calculated as

$$d_n^p = \frac{D_n}{C_n}, \tag{4}$$

and now, we obtain the value of d_n^p through the computation.

3.1.2. Local Communication Model

In the local communication model, the local communication latency in device n is denoted as l_{n}. The speed from the CPU's RAM to the GPU's VRAM is denoted as s_n^{cg}, the speed from the disk to the CPU's RAM is denoted as s_n^{dc}, the latency from the CPU's RAM to the GPU's VRAM is denoted as l_n^{cg}, the latency from the disk to the CPU's RAM is denoted as l_n^{dc}, the latency from the disk to the GPU's VRAM is denoted as l_n^{dg}, and the model weight for one layer is denoted as w_n.

The latency from the CPU's RAM to the GPU's VRAM l_n^{cg} is calculated as

$$l_n^{cg} = \frac{w_n}{s_n^{cg}}, \tag{5}$$

the latency from the disk to the CPU's RAM l_n^{dc} for w is calculated as

$$l_n^{dc} = \frac{w_n}{s_n^{dc}}, \tag{6}$$

when offloading w into a disk, they need to be first loaded into the CPU's RAM from the disk, then from the CPU's RAM to the GPU's VRAM for computation. The latency from the disk to the GPU's VRAM l^{dg} for w is calculated as

$$l_n^{dg} = \frac{w_n}{s_n^{cg}} + \frac{w_n}{s_n^{dc}}, \tag{7}$$

and in an IoMT device, we denote by x the number of model layers offloading in the CPU's RAM in devices and denote by y the number of model layers offloading in the disk in devices. According to Equations (5)–(7), we can calculate l_n^{loc} as

$$l_n^{loc} = \frac{(x_n + y_n)w_n}{s_n^{cg}} + \frac{y_n w_n}{s_n^{dc}}, \tag{8}$$

and now, we obtain the value of l_n^{loc} through the computation.

3.1.3. Total Communication Latency

In this section, we calculate the total communication latency. The total communication latency is denoted as l^{total} among the set of IoMT devices $\mathcal{N}$.

In the interconnection communication model, different layers are distributed across various devices. In the beginning, device i needs to send a token to the device that contains the first layer of the model, which is device 1. Then, it is required to transmit the token from the first computation device to the last computation device, which is device N. Finally, the device results are transmitted back to device i. To calculate l_n^{com} among all devices, we calculate the sum l_n^{com} of all devices in $\mathcal{N}$ and add the l_{i1}^{com} and l_{Ni}^{com}. In the local communication model, there are N devices; we calculate the sum l_n^{loc} of all devices in $\mathcal{N}$.

According to Equations (1) and (8), the total communication latency can be calculated as

$$l^{total} = \sum_{n \in \mathcal{N}}^{N} l_n^{loc} + \sum_{n \in \mathcal{N}}^{N-1} l_n^{com} + l_{i1}^{com} + l_{Ni}^{com}, \tag{9}$$

and now, we obtain the value of l^{total} through the computation.

3.2. MedMixtral 8x7B

With the increasing demands in healthcare, both patients and physicians are turning to medical Q&A services more frequently. Patients seek these services for timely advice, while physicians utilize them to manage their mounting workloads. Utilizing LLMs offers an approach to tackle this issue, given their capability to comprehend human language and facilitate Q&A interactions akin to those between individuals.

LLMs can specifically leverage a wealth of knowledge from the medical literature, databases, and clinical records, effectively processing and synthesizing this vast amount of information. As a result, LLMs can more accurately assess a patient's condition. For patients, an LLM can provide a preliminary diagnosis, and for physicians, it can reduce the risks of misdiagnosis. Furthermore, LLMs can provide patients with easily comprehensible medical explanations, treatment alternatives, and preventive measures. Therefore, LLMs can enhance patients' health literacy, increase patients' engagement, and subsequently, augment the public's foundational medical knowledge.

As depicted in Figure 2, we fine-tune an LLM based on the MoE architecture, with the pre-trained model being Mixtral 8x7B and the dataset named HealthCareMagic 8x7B. The MoE model is deployed on both memory and disk. We evaluate its performance on iCliniq. We choose to use the Mixtral 8x7b model due to its high performance. Mixtral 8x7b adopts a sparse mixture-of-experts (SMoE) architecture [43], functioning as a decoder-only model, where feedforward blocks are selected from a set of eight distinct parameter groups. For each token in the layers, the routers select two of these experts to deal with the token and aggregate outputs. The SMoE architecture enhances the model's parameters while reducing costs and latency by using only a fraction of the total parameters.

Specifically, Mixtral 8x7B has a total of 46.7B parameters. Compared to the GPT3.5 base model, Mixtral demonstrates comparable or superior performance across most benchmarks. As an SMoE model, it retains the strengths of a traditional MoE model while capitalizing on the advantages offered by an SMoE architecture.

We use the HealthCareMagic-100k dataset to fine-tune Mixtral 8x7B for medical-patient dialogue. In these conversations, patients often describe their conditions in non-standard ways. Self-made medical datasets can overuse specialized terminology and potentially introduce biased diagnoses, reducing data usefulness. Therefore, it is crucial to use real-world medical-patient dialogues from trusted sources to maintain data quality and reliability. The HealthCareMagic-100k dataset includes around 100,000 authentic doctor–patient dialogues from HealthCareMagic [44], with identifiers deleted and grammar errors corrected. This makes the HealthCareMagic-100k dataset an effective, information-rich, and comprehensive dataset, making it an excellent fit for medical Q&As.

The MedMixtral 8x7B model, which is fine-tuned on the HealthCareMagic-100k dataset, represents a novel medical LLM, enabling deployment on IoMT devices. By fine-tuning the pre-trained LLM with medical Q&A datasets, the MedMixtral 8x7B LLM can address medical inquiries and show exceptional performance in medical Q&A tasks.

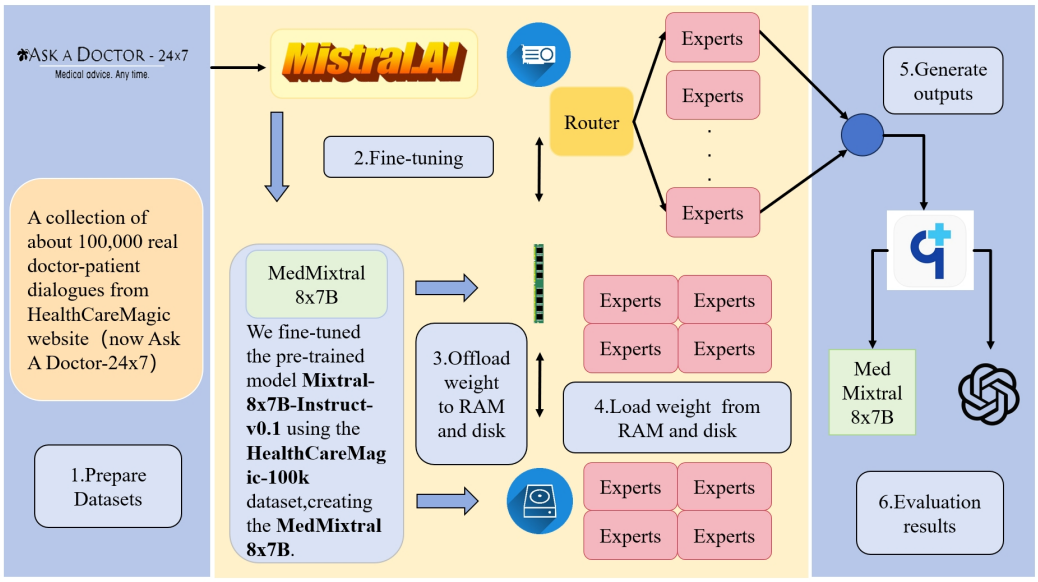

Figure 2. This is the MedMixtral 8x7B workflow. First, we prepare a dataset of about 100,000 samples from HealthCareMagic. Then, we fine-tune the Mixtral 8x7B model. After that, we design a strategy to offload the weights to both RAM and disk, aiming to alleviate the resource strain on IoMT deployments of LLMs. We obtain medical advice by asking the MedMixtral 8x7B model, and finally, evaluate our model's performance on iCliniq.

MedMixtral 8x7B demonstrates reduced bias and enhanced accuracy in medical Q&A tasks compared to traditional transformer-based LLMs. Figure 3 shows a medical Q&A example using MedMixtral 8x7B in the IoMT. The following paragraphs show some real and comprehensive answers of MedMixtral 8x7B to medical Q&As.

- Question1: Hello doctor, We have a 9-month-old baby girl. She has five brown marks, two of them are 2 cm size and the other three of 0.5 cm. They are irregular in shape. Initial two marks appeared when she was 3 months old and now when she is 9 months three more for added that are comparatively light and 0.5 cm of size. Are these birthmarks? Will they fade with the time? Do we need to take any precautions so that they do not increase or get added further? Can we also provide any medication to cure them?
- Answer1: Hello, The brown marks you are describing are commonly known as birthmarks. The two larger ones that appeared when your baby was three months old are likely to be called "cafe-au-lait" spots, while the smaller, lighter ones may be called freckles. Cafe-au-lait spots are harmless and may fade slightly with time, but they do not typically disappear completely. Freckles, on the other hand, may darken with sun exposure but can also be lightened with topical treatments. While these birthmarks are generally not a cause for concern, it is always a good idea to monitor any changes in their size, shape, or color. If you notice any new or unusual growths, or if the existing birthmarks become significantly larger or darker, it is recommended to consult a healthcare professional. In terms of precautions, it is important to protect your baby's skin from the sun, as UV radiation can darken the birthmarks and make them more noticeable. You can do this by dressing your baby in lightweight, long-sleeved clothing and a wide-brimmed hat, and by using a broad-spectrum sunscreen with an SPF of at least 30 on any exposed skin. As for medication, there is currently no cure or effective treatment for birthmarks. However, if the birthmarks are causing any discomfort, such as itching or irritation, your healthcare provider may recommend a topical cream or ointment to alleviate the symptoms. I hope this information is helpful. Please let me know if you have any further questions or concerns.
- Question2: Hello doctor, I have been going to a dentist about a tooth that is sore. They started a root canal but then did not finish. I have gone back several times and every

time they open up the tooth and insert medicine. My tooth is in agony right now. I am just wondering if this is a standard procedure during a root canal? Should not the dentist remove the nerves so I am not in so much pain? Or is it the standard procedure to fight the infection first and then remove the nerves?

- Answer2: Hello, During a root canal procedure, the dentist typically removes the nerve tissue first in order to alleviate the pain caused by the infection. However, in some cases where the infection is severe, the dentist may choose to administer medication and apply a temporary filling to the tooth to eliminate the infection before completing the root canal. This is often done to make the procedure more manageable for the patient and to ensure the best possible outcome. The fact that your dentist has been repeatedly opening your tooth and applying medication suggests that they are trying to eliminate the infection before completing the root canal. While this approach can be somewhat uncomfortable, it is not uncommon and is often necessary to ensure the success of the procedure. I would recommend speaking with your dentist about your concerns and asking for more information about the approach they are taking. They may be able to provide you with additional pain relief measures or offer further insight into the procedure and what to expect.
- Question3: Hello doctor, I have been diagnosed with mild hereditary spherocytosis. My eyes still yellow. Is it unlikely hematologist who diagnosed me three years ago misdiagnosed? I do not doubt he knows his stuff. More about me getting worried. I lost weight a couple of months ago. I went very yellow in the eyes. I have always felt tired. I have had gallbladder out four years ago, I bruise easily, I do get little red skin dots at times, menstrual cycles can be irregular and spleen slightly enlarged.
- Answer3: Hello, While it is unlikely that your hematologist misdiagnosed you, the symptoms you have described could be indicative of other conditions such as liver disease or a more severe form of hereditary spherocytosis. The jaundice you mentioned, along with weight loss, fatigue, easy bruising, and enlarged spleen, are all potential symptoms of these conditions. It would be best to consult with your hematologist and possibly a gastroenterologist or liver specialist to rule out any other potential causes for your symptoms. They may want to perform additional tests, such as a liver function test or a bone marrow biopsy, to further evaluate your condition. It is important to remember that this advice is intended to be general in nature, and specific causes may not apply to your situation. Always consult with a healthcare professional for an accurate diagnosis. I hope this information helps, and please let me know if you have any further questions or concerns.
- Question4: Good evening, doctor. I have a severe toothache.It hurts so bad.It is difficult for me to eat.The pain is like a sharp knife cutting my teeth.What is the cause of it?
- Answer4: Good evening, I'm sorry to hear that you're experiencing tooth pain. The description you provided suggests that the pain is severe and sharp, which could indicate a few possible causes. The most common cause of tooth pain is tooth decay, which can cause a sharp pain when eating or drinking something sweet, hot or cold. Another possible cause is an infection or abscess in the tooth or gum, which can cause a persistent, throbbing pain. In some cases, tooth pain can also be caused by gum disease, injury, or a cracked tooth. It is important to see a dentist as soon as possible to determine the cause of the pain and to receive appropriate treatment. In the meantime, you can try taking over-the-counter pain relievers to help manage the pain. However, it is not recommended to apply pain relievers directly to the tooth or gums, as this can burn the gum tissue.

In the model fine-tuning process, we employed the LoRA fine-tuning method [45]. LoRA offers a fine-tuning approach that consumes less VRAM and memory while achieving performance close to full fine-tuning. In the fine-tuning setup, we set warmup steps to 0.03, max steps to 1000, learning rate to 2×10^{-4}, and logging steps to 1.

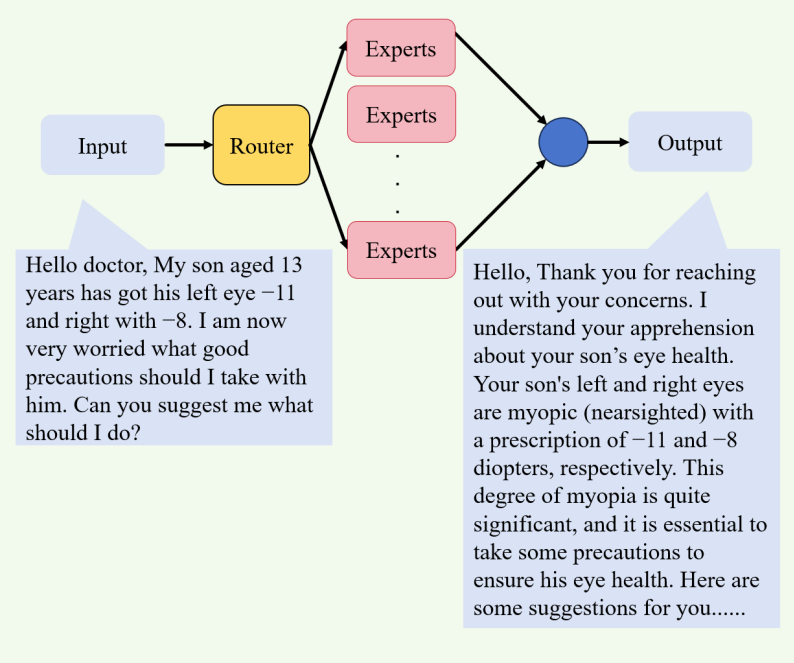

Figure 3. An instance of the process of MedMixtral 8x7B generating medical answers. After the input tokens enter the model, it is routed to the experts by the router. Then, the experts generate responses based on the input tokens.

3.3. Efficient Inference Offloading

The increasing adoption of LLMs is spurring demand for innovative LLM architectures with superior performance attributes. For instance, SMoE is one of the emergent architectures, where only specific model layers are activated for any given input, making it particularly useful for tasks such as NLP. In SMoE, this feature allows LLMs to generate tokens faster than before, even though it leads to an increase in model size due to the integration of multiple experts. Therefore, deploying high-performance LLMs demands considerable VRAM and high-performance GPUs to ensure optimal operation. We propose a novel strategy called efficient inference offloading to address the challenge of deploying MedMixtral 8x7B on IoMT devices.

Efficient Inference Offloading Algorithm

For the design of efficient inference offloading algorithms, we employ the least recently used (LRU) cache strategy [3] to dynamically adjust the number of experts per layer, based on VRAM size. In instances of limited VRAM availability, we augment the number of experts, whereas we decrease it when VRAM is abundant. Upon loading all experts of the current layer, we start the expert loading based on the 1–2 most probable experts derived from the inference results. These newly loaded experts do not replace any existing experts in the cache; however, if used in the inference of the subsequent layer, they replace the least recently used expert in the cache.

Meanwhile, we introduce an efficient strategy for loading model weights. First, an empty model framework is loaded into the CPU's RAM, minimizing the CPU's RAM consumption. Then, model weights are loaded into the CPU's RAM and stored on the disk as the configuration. The loaded weights are moved from the CPU's RAM to the disk, then the checkpoint is removed from the CPU's RAM. When loading the model weights, the loaded weights are moved from the disk to the CPU's RAM. Hooks are attached to each weight of the model, enabling the transfer of weights from the CPU's RAM to the GPU when needed, and back to the CPU's RAM after completing the associated computations.

We chose to offload certain parameters to the disk due to constraints in memory availability. Our offloading strategy prioritizes memory utilization, resorting to disk offloading

only when memory resources are insufficient given the considerable size of parameters in LLMs. In SMoE, this feature allows LLMs to generate tokens faster than before, even though it leads to an increase in model size due to the integration of multiple experts. We paid particular attention to the performance of IoMT devices. Considering the limited memory capacity of IoMT devices in current scenarios, we specifically consider the option of offloading to the disk. Because we can offload to the disk, our flexible offloading strategy enables IoMT devices to support larger models without compromising performance.

4. Results

In this section, we first elaborate on our experimental environment by using a 36 vCPU AMD EPYC 9754 128-core Processor (AMD, Santa Clara, CA, USA) and NVIDIA GTX 3090 x2 (NVIDIA, Santa Clara, CA, USA).

4.1. MedMixtral 8x7B

MedMixtral 8x7B is a medical LLM designed for deployment on IoMT devices. By fine-tuning Mixtral 8x7B with medical datasets, the MedMixtral 8x7B proficiently addresses medical questions while demonstrating exceptional performance. The MedMixtral 8x7B consists of multiple experts, similar to conventional medical practice with multiple experts, which enhances the accuracy of medical diagnostics.

Comparative evaluations are conducted among our MedMixtral 8x7B, ChatGPT, Llama3 8B, and the original Mixtral 8x7b model. By utilizing the 4-bit quantized Mixtral 8x7B and MedMixtral 8x7B in this experiment, we observe from Table 1 that MedMixtral 8x7B consistently outperforms them in terms of Q&A accuracy.

The results indicate that our MedMixtral 8x7B shows significant improvements in the precision, recall, and F1 score metrics compared to the original Mixtral 8x7B metrics. Additionally, it slightly outperforms ChatGPT and Llama3 8B in all metrics. This demonstrates that our MedMixtral 8x7B exhibits superior performance in medical Q&As.

Table 1. Quantitative comparison among ChatGPT, Mixtral 8x7B, and MedMixtral 8x7B. According to the test results, our model MedMixtral 8x7B, which is fine-tuned on Mixtral 8x7B, outperforms other models in terms of precision, recall, and F1 score when answering medical questions on the test set. This indicates that our model has an advantage in the medical Q&A domain compared to Mixtral 8x7B, Llama3 8B, and ChatGPT.

	Precision	**Recall**	**F1 Score**
ChatGPT	0.837	0.8445	0.8406
Mixtral 8x7B	0.821	0.8434	0.8320
MedMixtral 8x7B	0.838	0.8447	0.8413
Llama 8B	0.799	0.8384	0.8179

4.2. Offloading Strategy

Figure 4 illustrates the occupied VRAM capacity as the number of offloaded experts changes. When no expert has been offloaded, the quantized Mixtral 8x7B model can be deployed with 18.4 GB VRAM. Conversely, in original offloading, it requires 20.1 GB VRAM. With two experts offloaded, the quantized Mixtral 8x7B model can be deployed with 14.5 GB VRAM, whereas in original offloading it demands nearly 16.3 GB VRAM. Upon offloading four experts, the quantized Mixtral 8x7B model can be deployed with 10.1 GB VRAM, whereas in original offloading it demands 11.9 GB, close to 12 GB VRAM. Furthermore, the model weights are loaded into both the CPU's RAM and disk, a strategy that conserves a certain amount of VRAM capacity while expanding the loadable model size.

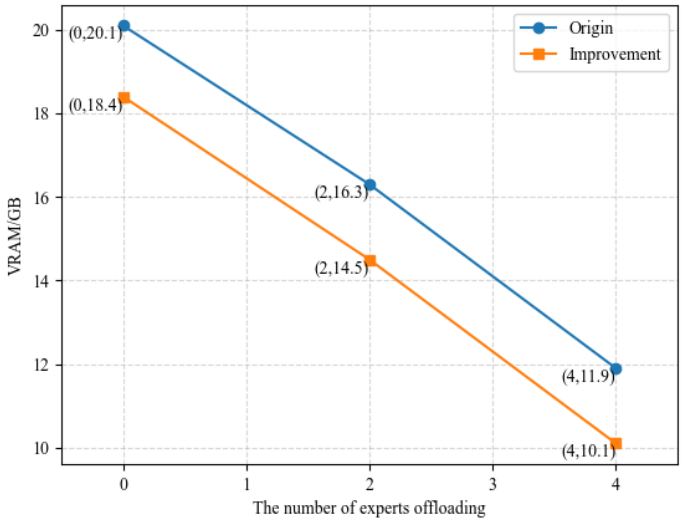

Figure 4. This is the trend in changes in VRAM usage with the number of offloading experts. Our strategy requires less VRAM capacity than the strategy method across all number of offloading experts scenarios.

4.3. Communication Latency

We conduct experiments to evaluate the interconnection communication latency associated with transferring intermediate parameters. In the experiment, we consider the communication latency l_n^{com} equals d_n^p and test its value.

Initially, we consider the interconnect via an SSH connection. The measured average l_n^{com} is recorded at 0.1039 s. Despite SSH having certain latency, it still has the advantage of facilitating distributed computing, upon network availability.

To reduce network latency, wireless networking emerges as a viable option for interconnection. In the test, the temperature is 290 K, and the B_n is 160 MHz. We measure an S_n of $10^{-3.90}$ mW. According to Equation (2), we calculate an ω_n of $10^{-9.19}$ mW in wireless networks. According to Equation (2), the SNR_n can be computed as $10^{5.29}$. After obtaining the SNR_n, according to Equation (3), we can calculate C_n as 2.796×10^9 bit/s. Converting 2.796×10^9 bits per second to megabytes per second gives 349.5 MB/s. The D_n is 64 bytes, according to Equation (4), we can calculate the minimum d_n^p as 1.748×10^{-7} s. Based on the results, we ascertain that the observed value of l_n^{com} is less than 0.001 s. The experiment meets expectations. Consequently, it can be inferred that the influence of using wireless on LLM is insignificant.

After testing interconnection communication latency, we test the local communication latency on the local device. We record the value of s_n^{dc} as 2.135 GBps and record the value of s_n^{cg} as 12.46 GBps. For the quantized LLM weights in the LLM, each layer is approximately 0.55 GB.

If the weight only loads from the CPU's RAM, according to Equation (5), the l_n^{cg} for each layer is approximately 0.044 s. However, if the weight loads from the disk, according to Equation (6), the l_n^{dc} for each layer is approximately 0.26 s. According to Equation (7), the l_n^{dg} for each layer is approximately 0.304 s. We assume that in the device, eight layers offload to CPU's RAM, and eight layers offload to disk, according to Equation (8), the l_n^{loc} can be calculated as 2.784 s. It is evident that the disk speed significantly influences the inference speed.

We find that opting for high-performance disks can significantly reduce latency. After using the high-performance NVMe drive, we boost the s_n^{dc} to 3.166 GBps. According to Equation (6), the l_n^{dc} is approximately calculated as 0.17 s. To calculate performance

improvement, we denote the improvement performance as $Improvement$, while the original latency is denoted as $Original$ and $Improved$ is the improved latency. The formula is

$$Improvement = \left(\frac{Original - Improved}{Original} \right) \times 100\%. \quad (10)$$

According to Equation (10), we can calculate that the performance improvement is approximately 32%.

Similarly, loading the majority of the model weights into the CPU's RAM leads to a substantial reduction in latency. However, in terms of the model's performance, the impact is only on latency and does not affect its inference capabilities.

Finally, we can calculate the l^{total}. We assume there are 32 IoMT devices similar to our device. Each device has one layer in the model. We use SSH connections and offload the weights to disk. According to Equation (9), we can approximately calculate the l^{total} as 13.16 s. If we offload the model weights to the CPU's RAM and use SSH connections, we can approximately calculate the l^{total} as 4.837 s. If we offload the model weights to the CPU's RAM and use the wireless connection, we can approximately calculate the l^{total} as 1.408 s.

5. Discussion and Limitations

The results demonstrate that our MedMixtral 8x7B, fine-tuned on the HealthCareMagic-100k dataset, excels in answering medical questions.

The robust performance of the MedMixtral 8x7B model is largely attributed to the extensive medical Q&A data provided by the HealthCareMagic-100k fine-tuning dataset. However, the model's remarkable performance cannot be solely attributed to its comprehensive learning of the dataset. The superior performance of MedMixtral 8x7B may also stem from the unique advantages of the MoE architecture in addressing medical Q&A issues. During the learning process, MedMixtral 8x7B, with its multiple experts, allows different experts to acquire diverse aspects of medical knowledge. When answering medical questions, the gate evaluates the input query and selects two suitable experts to respond. This method, which involves choosing a few experts rather than combining the opinions of all experts, reduces the model's bias and enhances accuracy.

Our model, fine-tuned on the HealthCareMagic-100k dataset, surpasses traditional models such as ChatGPT and Llama3 in terms of precision, recall, and F1 score. Moreover, MedMixtral 8x7B is compatible with our proposed efficient inference offloading architecture. Our efficient inference offloading algorithm dynamically adjusts the number of experts per layer based on available VRAM, thereby reducing resource consumption. By employing the LRU cache strategy and introducing model weight offloading techniques, MedMixtral 8x7B exhibits lower VRAM usage during deployment, making it a potential candidate for IoMT devices. Compared to other large LLMs, MedMixtral 8x7B offers superior or comparable medical Q&A capabilities with reduced resource consumption—utilizing VRAM typically required by models under 10B in size. Compared to other small LLMs deployable on IoMT devices, MedMixtral 8x7B provides a significant performance advantage in medical Q&As. Additionally, deploying MedMixtral 8x7B locally can effectively protect user privacy and mitigate network security risks.

In addition to addressing resource requirements, we examined the impact of latency on LLM inference speed. Through investigating local and interconnection communication models, we identified key factors affecting latency, including device interconnection methods, offloading locations, and disk speeds. We propose strategies to reduce latency, such as enhancing disk speed and utilizing wireless communication instead of SSH connections.

Our study highlights the potential of MedMixtral 8x7B in answering medical questions. Below, we discuss the limitations and outline directions for future research.

5.1. Hallucinations in LLMs within Medical Settings

Despite the promising results, our manual evaluation on consumer medical Q&A datasets indicates that the accuracy of existing LLMs remains insufficient. Models like ChatGPT and MedMixtral 8x7B may generate inaccurate answers in the medical domain, posing significant safety risks. In the future, we aim to enhance accuracy and safety by integrating techniques such as chain of thought (CoT), thereby narrowing the gap between these models and real-world clinicians, and facilitating quicker clinical adoption.

5.2. Expansion of the HealthCareMagic-100k Dataset

The HealthCareMagic-100k dataset encompasses a variety of medical Q&As from diverse sources, but it is not exhaustive. We plan to expand this dataset to include a broader range of medical Q&As. During this expansion, preprocessing data obtained from patients presents a challenge. Compared to multiple-choice tasks, preprocessing Q&A tasks is more complex. The varied tones and expressions in Q&A pairs affect LLM fine-tuning, requiring experimentation to determine which types of Q&As yield the best learning outcomes. We aim to preprocess the data accordingly for optimal results. Additionally, the HealthCareMagic-100k dataset is purely in English. In the future, we will seek to incorporate multilingual Q&As and evaluate multilingual capabilities.

5.3. Improving Evaluation Methods

We evaluated model performance on the iCliniq dataset, using metrics such as precision, recall, and F1 score, with BERTscore as our evaluation method. While BERTscore's advantage lies in its ability to assess answers' similarity to human responses, offering a more realistic measure than conventional test questions, its drawback is that it only measures similarity and not specific accuracy. Given the critical nature of medical Q&As, where errors can have severe consequences, we plan to incorporate additional evaluation methods, such as multiple-choice questions, to enhance the fairness and comprehensiveness of model assessments.

6. Conclusions

Through development and fine-tuning based on the HealthCareMagic-100k dataset, our MedMixtral 8x7B model has emerged as a new solution deployable in the medical Q&A domain, particularly on consumer devices such as IoMT devices. Its outstanding performance is attributed not only to the richness of the fine-tuned dataset but also to the utilization of the MoE architecture in the base model, which incorporates multiple experts to reduce bias and enhance accuracy.

Moreover, the integration of MedMixtral 8x7B with efficient inference offloading architectures makes it an excellent LLM deployable on medical IoMT devices. By dynamically adjusting resource allocation based on available VRAM and employing techniques like LRU cache and model weight offloading, MedMixtral 8x7B demonstrates remarkable medical Q&A capabilities while minimizing resource consumption. This gives it a significant advantage over larger language models and smaller deployable alternatives, while also maximizing user privacy protection and mitigating network attack concerns on IoMT devices.

Finally, strategies for reducing latency were explored, indicating that optimizing device interconnection methods, optimizing weight offloading locations, and enhancing disk speed all contribute to improving the overall efficiency and performance of MedMixtral 8x7B, enabling faster and more reliable inference. This positions it as one of the most suitable medical Q&A LLMs for deployment on IoMT devices.

Author Contributions: In this project, X.Y. and W.K. contributed to resource allocation, methodology development, software implementation, data validation, formal analysis, and original draft writing. X.Y. also contributed to supervision, and funding acquisition. Z.L. and M.X. contributed to conceptu-

alization, formal analysis, investigation, and visualization. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the National Natural Science Foundation of China (62371116), in part by the Science and Technology Project of Hebei Province Education Department (ZD2022164), and in part by the Project of Hebei Key Laboratory of Software Engineering (22567637H).

Data Availability Statement: Data are contained within the article.

Acknowledgments: We would like to thank the editors and the anonymous reviewers for their insightful comments and constructive suggestions.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

Symbol	Description in Device n
B_n	Channel bandwidth
C_n	Channel capacity
d_n^p	Propagation delay
d_n^{pr}	Processing delay
d_n^q	Queueing delay
d_n^w	Waiting delay
D_n	Data volume
k	Boltzmann constant
l_n^{cg}	Latency from the CPU's RAM to the GPU's VRAM
l_n^{com}	Interconnection communication latency
l_n^{dc}	Latency from the disk to the CPU's RAM
l_n^{dg}	Latency from the disk to the GPU's VRAM
l_n^{loc}	Local communication latency
$\mathcal{N}$	Set of IoMT devices
w_n	Model weight for one layer
s_n^{cg}	Speed from the CPU's RAM to the GPU's VRAM
s_n^{dc}	Speed from the disk to the CPU's RAM
S_n	Signal power
SNR_n	Signal-to-noise ratio
T	Absolute temperature in kelvin
l^{total}	Total communication latency
x_n	The number of model layers in device n
y_n	The number of model layers offloading in the disk in device n
ω_n	White Gaussian noise

References

1. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
2. Amin, S.U.; Hossain, M.S. Edge intelligence and Internet of Things in healthcare: A survey. *IEEE Access* **2020**, *9*, 45–59. [CrossRef]
3. Xu, M.; Niyato, D.; Zhang, H.; Kang, J.; Xiong, Z.; Mao, S.; Han, Z. Cached Model-as-a-Resource: Provisioning Large Language Model Agents for Edge Intelligence in Space-air-ground Integrated Networks. *arXiv* **2024**, arXiv:2403.05826.
4. Xu, M.; Dusit, N.; Kang, J.; Xiong, Z.; Mao, S.; Han, Z.; Kim, D.I.; Letaief, K.B. When large language model agents meet 6g networks: Perception, grounding, and alignment. *arXiv* **2024**, arXiv:2401.07764.
5. Jiang, A.Q.; Sablayrolles, A.; Roux, A.; Mensch, A.; Savary, B.; Bamford, C.; Chaplot, D.S.; de las Casas, D.; Hanna, E.B.; Bressand, F.; et al. Mixtral of experts. *arXiv* **2024**, arXiv:2401.04088.
6. Eliseev, A.; Mazur, D. Fast inference of mixture-of-experts language models with offloading. *arXiv* **2023**, arXiv:2312.17238.
7. Gugger, S.; Debut, L.; Wolf, T.; Schmid, P.; Mueller, Z.; Mangrulkar, S.; Sun, M.; Bossan, B. Accelerate: Training and Inference at Scale Made Simple, Efficient and Adaptable. 2022. Available online: https://github.com/huggingface/accelerate (accessed on 6 March 2021).
8. Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. Improving Language Understanding by Generative Pre-Training. 2018. Available online: https://www.mikecaptain.com/resources/pdf/GPT-1.pdf (accessed on 11 June 2018).

9. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. *OpenAI Blog* **2019**, *1*, 9.
10. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
11. Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. Training language models to follow instructions with human feedback. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 27730–27744.
12. Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F.L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. Gpt-4 technical report. *arXiv* **2023**, arXiv:2303.08774.
13. Zhao, W.X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; Hou, Y.; Min, Y.; Zhang, B.; Zhang, J.; Dong, Z.; et al. A survey of large language models. *arXiv* **2023**, arXiv:2303.18223.
14. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
15. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
16. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **2020**, *21*, 1–67.
17. Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. LLaMA: Open and Efficient Foundation Language Models. *arXiv* **2023**, arXiv:2302.13971.
18. Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv* **2023**, arXiv:2307.09288.
19. Zhang, S.; Roller, S.; Goyal, N.; Artetxe, M.; Chen, M.; Chen, S.; Dewan, C.; Diab, M.; Li, X.; Lin, X.V.; et al. Opt: Open pre-trained transformer language models. *arXiv* **2022**, arXiv:2205.01068.
20. Zhang, Z.; Han, X.; Zhou, H.; Ke, P.; Gu, Y.; Ye, D.; Qin, Y.; Su, Y.; Ji, H.; Guan, J.; et al. CPM: A large-scale generative Chinese pre-trained language model. *AI Open* **2021**, *2*, 93–99. [CrossRef]
21. Zhang, Z.; Gu, Y.; Han, X.; Chen, S.; Xiao, C.; Sun, Z.; Yao, Y.; Qi, F.; Guan, J.; Ke, P.; et al. Cpm-2: Large-scale cost-effective pre-trained language models. *AI Open* **2021**, *2*, 216–224. [CrossRef]
22. Nijkamp, E.; Pang, B.; Hayashi, H.; Tu, L.; Wang, H.; Zhou, Y.; Savarese, S.; Xiong, C. Codegen: An open large language model for code with multi-turn program synthesis. *arXiv* **2022**, arXiv:2203.13474.
23. Le Scao, T.; Fan, A.; Akiki, C.; Pavlick, E.; Ilić, S.; Hesslow, D.; Castagné, R.; Luccioni, A.S.; Yvon, F.; Gallé, M.; et al. Bloom: A 176b-Parameter Open-Access Multilingual Language Model. 2023. Available online: https://inria.hal.science/hal-03850124/ (accessed on 20 November 2023).
24. Zeng, A.; Liu, X.; Du, Z.; Wang, Z.; Lai, H.; Ding, M.; Yang, Z.; Xu, Y.; Zheng, W.; Xia, X.; et al. Glm-130b: An open bilingual pre-trained model. *arXiv* **2022**, arXiv:2210.02414.
25. Luo, R.; Sun, L.; Xia, Y.; Qin, T.; Zhang, S.; Poon, H.; Liu, T.Y. BioGPT: Generative pre-trained transformer for biomedical text generation and mining. *Briefings Bioinform.* **2022**, *23*, bbac409. [CrossRef] [PubMed]
26. Singhal, K.; Azizi, S.; Tu, T.; Mahdavi, S.S.; Wei, J.; Chung, H.W.; Scales, N.; Tanwani, A.; Cole-Lewis, H.; Pfohl, S.; et al. Large language models encode clinical knowledge. *arXiv* **2022**, arXiv:2212.13138.
27. Singhal, K.; Tu, T.; Gottweis, J.; Sayres, R.; Wulczyn, E.; Hou, L.; Clark, K.; Pfohl, S.; Cole-Lewis, H.; Neal, D.; et al. Towards expert-level medical question answering with large language models. *arXiv* **2023**, arXiv:2305.09617.
28. Rasley, J.; Rajbhandari, S.; Ruwase, O.; He, Y. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, 6–10 July 2020; pp. 3505–3506.
29. Shoeybi, M.; Patwary, M.; Puri, R.; LeGresley, P.; Casper, J.; Catanzaro, B. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv* **2019**, arXiv:1909.08053.
30. Kwon, W.; Li, Z.; Zhuang, S.; Sheng, Y.; Zheng, L.; Yu, C.H.; Gonzalez, J.; Zhang, H.; Stoica, I. Efficient memory management for large language model serving with pagedattention. In Proceedings of the 29th Symposium on Operating Systems Principles, Koblenz, Germany, 23–26 October 2023; pp. 611–626.
31. Zhao, Y.; Gu, A.; Varma, R.; Luo, L.; Huang, C.C.; Xu, M.; Wright, L.; Shojanazeri, H.; Ott, M.; Shleifer, S.; et al. Pytorch fsdp: Experiences on scaling fully sharded data parallel. *arXiv* **2023**, arXiv:2304.11277.
32. Han, P.; Wang, S.; Leung, K.K. Adaptive gradient sparsification for efficient federated learning: An online learning approach. In Proceedings of the 2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS), Singapore, 29 November–1 December 2020; pp. 300–310.
33. Han, P.; Wang, S.; Jiao, Y.; Huang, J. Federated Learning While Providing Model as a Service: Joint Training and Inference Optimization. *arXiv* **2023**, arXiv:2312.12863.
34. Piovesan, N.; López-Pérez, D.; De Domenico, A.; Geng, X.; Bao, H.; Debbah, M. Machine learning and analytical power consumption models for 5G base stations. *IEEE Commun. Mag.* **2022**, *60*, 56–62. [CrossRef]
35. Ayed, F.; De Domenico, A.; Garcia-Rodriguez, A.; López-Pérez, D. Accordion: A communication-aware machine learning framework for next generation networks. *IEEE Commun. Mag.* **2023**, *61*, 104–110. [CrossRef]

36. Du, Y.; Liew, S.C.; Chen, K.; Shao, Y. The power of large language models for wireless communication system development: A case study on fpga platforms. *arXiv* **2023**, arXiv:2307.07319.
37. Bariah, L.; Zhao, Q.; Zou, H.; Tian, Y.; Bader, F.; Debbah, M. Large language models for telecom: The next big thing? *arXiv* **2023**, arXiv:2306.10249.
38. Bariah, L.; Zou, H.; Zhao, Q.; Mouhouche, B.; Bader, F.; Debbah, M. Understanding telecom language through large language models. In Proceedings of the GLOBECOM 2023—2023 IEEE Global Communications Conference, Kuala Lumpur, Malaysia, 4–8 December 2023; pp. 6542–6547.
39. Soman, S.; HG, R. Observations on LLMs for telecom domain: Capabilities and limitations. *arXiv* **2023**, arXiv:2305.13102.
40. Zhang, T.; Xu, C.; Lian, Y.; Tian, H.; Kang, J.; Kuang, X.; Niyato, D. When Moving Target Defense Meets Attack Prediction in Digital Twins: A Convolutional and Hierarchical Reinforcement Learning Approach. *IEEE J. Sel. Areas Commun.* **2023**, *41*, 3293–3305. [CrossRef]
41. Zhang, T.; Xu, C.; Shen, J.; Kuang, X.; Grieco, L.A. How to Disturb Network Reconnaissance: A Moving Target Defense Approach Based on Deep Reinforcement Learning. *IEEE Trans. Inf. Forensics Secur.* **2023**, *18*, 5735–5748. [CrossRef]
42. Zhang, T.; Xu, C.; Zou, P.; Tian, H.; Kuang, X.; Yang, S.; Zhong, L.; Niyato, D. How to mitigate DDoS intelligently in SD-IoV: A moving target defense approach. *IEEE Trans. Ind. Inform.* **2022**, *19*, 1097–1106. [CrossRef]
43. Fedus, W.; Zoph, B.; Shazeer, N. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *J. Mach. Learn. Res.* **2022**, *23*, 1–39.
44. Li, Y.; Li, Z.; Zhang, K.; Dan, R.; Jiang, S.; Zhang, Y. ChatDoctor: A Medical Chat Model Fine-Tuned on a Large Language Model Meta-AI (LLaMA) Using Medical Domain Knowledge. *Cureus* **2023**, *15*, e40895. [CrossRef]
45. Hu, E.J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W. Lora: Low-rank adaptation of large language models. *arXiv* **2021**, arXiv:2106.09685.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.