Challenges in Adopting LLaMA: An Empirical Study of Discussions on Stack Overflow

Ramita Deeprom[1,*], Shiyu Yang[2], Yoshiki Higo[2], Morakot Choetkiertikul[1] and Chaiyong Ragkhitwetsagul[1]

[1]Faculty of Information and Communication Technology, Mahidol University, 999 Phuttamonthon 4 Road, Salaya, Nakhon Pathom 73170 THAILAND
[2]Graduate School of Information Science and Technology, Osaka University 1-5, Yamadaoka, Suita, Osaka, 565-0871, Japan

Abstract

LLaMA (Large Language Model Meta AI) has quickly gained traction among developers due to its wide-ranging applications and its capabilities to be integrated into software projects. As interest in LLaMA grows, discussions around it have surged on platforms like Stack Overflow. The developer community, with its collaborative nature, serves as a valuable source for studying LLaMA's quality, its emerging trends, and insights into its usage. Despite this growing attention, there has been no comprehensive study examining how the community interacts with and discusses LLaMA. This study addresses that gap by exploring conversations on Stack Overflow related to LLaMA and its quality, with the objective of identifying key themes and recurring patterns in these discussions. We systematically collected and analyzed 473 posts from Stack Overflow that contained the keyword "LLaMA" or were tagged accordingly. The analysis revealed that prominent topics of discussion include model configuration, error handling, and integration with other technologies. Furthermore, we identified frequent co-occurring tags, underscoring LLaMA's integration within the larger ecosystem of large language models and its interoperability with widely used frameworks, such as Python and Hugging Face Transformers. The findings highlight the complexity of working with LLaMA, especially in model configuration and fine-tuning, indicating a need for better resources, documentation, and community support. The study also suggests that future development should prioritize interoperability with popular machine-learning frameworks to improve the LLM's quality and to strengthen LLaMA's role in the AI ecosystem.

Keywords

LLaMA, Stack Overflow, Large Language Models' Quality

1. Introduction

The rapid advancements in artificial intelligence (AI) have revolutionized the field of technology, leading to the creation of powerful large language models (LLMs) that are transforming how developers and organizations approach problem-solving. One such model is Meta's LLaMA[1], an open-source LLM that has garnered substantial attention from the developer community [1]. Unlike many proprietary models, LLaMA offers developers the flexibility to fine-tune and customize the model for specific use cases, making it an attractive alternative for those who require more control and adaptability in their applications [1].

Recent studies have demonstrated LLaMA's superior performance in specific domain tasks, such as cheminformatics, where it has outperformed models like ChatGPT in tasks such as SMILES embeddings for predicting molecular properties and drug-drug interactions (DDI) [2]. This suggests that LLaMA is particularly effective in tasks that demand high degree of precision and domain-specific expertise, setting it apart from other LLMs. While models like ChatGPT, Bard, and Ernie may offer unique features such as real-time web access or higher computational efficiency, LLaMA stands out by providing a well-rounded balance across various criteria, making it suitable for a broader range of applications [3].

The growing interest in LLaMA is particularly evident on platforms like Stack Overflow (SO),[2] an online community where developers ask questions, share knowledge, and provide solutions related to software development and technology. SO has become one of the most widely used platforms for developers to collaborate, troubleshoot, and learn from each other, making it a rich source of information about real-world challenges and practical applications of various technologies. Studying SO is essential because it reflects the collective experiences and expertise of a global community of developers, providing valuable insights into the quality, common issues, and trends that arise with new technologies like LLaMA. By examining the discussions on SO, we can better understand not only the key themes and challenges developers face with LLaMA but also the broader context of its integration and adoption in various fields. This understanding is critical for identifying areas where additional support, documentation, or tools might be needed to improve the developer experience and further promote the effective use of LLaMA.

This study aims to address an initial gap by conducting an empirical analysis of Stack Overflow posts tagged with LLaMA to identify the predominant discussion topics related to its quality and adoption, and associated technologies. By employing keyword frequency analysis and categorizing the posts, this study seeks to answer two key research questions: (1) What are the main topics of discussion regarding LLaMA on Stack Overflow? and (2) What related themes emerge in these discussions? Through this initial analysis, we aim to provide early insights into the specific challenges developers face, the solutions they seek, and the broader implications for LLaMA's role within the AI ecosystem. The findings from this research study will serve as a foundation for a more comprehensive future study, contributing valuable insights to both practitioners and researchers as we further our understanding of LLaMA's use and integration within diverse technical environments.

The structure of this paper is as follows. Section 2 pro-

QuASoQ 2024: 12th International Workshop on Quantitative Approaches to Software Quality, December 03, 2024, Chongqing, China
*Corresponding author.
ramita.dep@student.mahidol.ac.th (R. Deeprom); yangsy@ist.osaka-u.ac.jp (S. Yang); higo@ist.osaka-u.ac.jp (Y. Higo); morakot.cho@mahidol.ac.th (M. Choetkiertikul); chaiyong.rag@mahidol.ac.th (C. Ragkhitwetsagul)
https://ysy-dlg.github.io/MyHomePage (S. Yang); https://sites.google.com/view/yhigo/home (Y. Higo); https://morakotch.wordpress.com/ (M. Choetkiertikul); https://cragkhit.github.io/ (C. Ragkhitwetsagul)

[1]https://github.com/meta-llama/llama

[2]https://stackoverflow.com/

vides the background and related work, detailing prior research on the adoption of large language models (LLMs) such as LLaMA and their application in real-world scenarios. The methodology employed in our research, including data collection and preprocessing techniques, is explained in Section 3. Section 4 presents the results of our empirical study, focusing on the analysis of Stack Overflow discussions to answer the research questions posed in this study. We then discuss the implications of our findings in Section 5, where we highlight the key challenges faced by developers when working with LLaMA and suggest potential improvements for future development. Finally, Section 6 concludes the paper and outlines potential avenues for future research, such as expanding the dataset and exploring more advanced stages of LLaMA adoption.

2. Background and Related Work

The rapid adoption of generative AI, particularly large language models (LLMs), has sparked significant interest in understanding how users are integrating these tools into their workflows. Previous research shows that many professionals increasingly rely on generative AI, such as ChatGPT and LLaMA, to solve problems traditionally addressed on platforms like Stack Overflow (SO) [4, 5]. This shift suggests a change in the problem-solving paradigm, where AI-generated solutions are becoming a first resort for many developers, streamlining the troubleshooting process and improving efficiency [4]. However, despite the growing reliance on AI, recent studies indicate that not all users are fully satisfied with AI-generated responses. Some developers still face challenges, particularly with complex technical issues, prompting them to seek human-based community support on platforms like SO [6, 5]. This highlights the limitations of AI models in delivering contextually accurate and reliable answers for more nuanced problems [7, 5].

LLaMA, an open-source LLM created by Meta, offers notable advantages that contribute to its rising popularity within the developer community. Released to the public in February 2023, with LLaMA 3.1 debuting in July 2024, the model has garnered over 300 million downloads globally, underscoring its widespread adoption [8]. Compared to ChatGPT, LLaMA is perceived as more complex to install and configure, yet its appeal lies in its ability to provide fine-tuned, context-specific outputs, making it particularly attractive to developers who require precision and control [8, 3]. Furthermore, LLaMA's enhanced security features and the ability to be hosted internally within organizations without the risk of leaking sensitive information make it a strong contender for enterprise use cases [3]. These characteristics reduce the risk of biased outputs, which is often a concern for beginners relying too heavily on AI-generated responses [3]. The model's open-source nature also allows for greater flexibility in integration and customization, offering experienced developers a robust tool for specialized applications [3, 2].

Studies have highlighted that LLaMA excels in certain domain-specific tasks, such as cheminformatics, where it outperforms ChatGPT in Simplified Molecular Input Line Entry System (SMILES) embeddings for molecular property and drug-drug interaction (DDI) predictions [2]. This superior performance suggests that LLaMA is well-suited for tasks that require high degree of precision and the handling of specific domain data, further distinguishing it from other LLMs. Similarly, comparative analyses have shown that while ChatGPT and other models like Bard and Ernie offer advantages in certain areas, such as real-time internet access or computational efficiency, LLaMA provides balanced performance across multiple criteria, making it a versatile tool for various applications [3].

Moreover, the performance of Llama 2 has been noted to exhibit minimal variation across different languages, offering consistency in sentiment analysis tasks. However, this consistency sometimes comes at the cost of skewing ratings towards positive sentiment, even in scenarios where more nuanced interpretations are required [9]. Furthermore, recent studies on job recommendations generated by LLaMA reveal both strengths and limitations. While LLaMA suggests a wider variety of professions compared to ChatGPT, its recommendations often include impractical or nonsensical roles, reflecting a trade-off between diversity and practicality [10]. This indicates the need for improved prompt engineering and bias mitigation in LLM applications to ensure fairer and more relevant outcomes across diverse user groups.

Several studies have leveraged Stack Overflow data to analyze trends within the developer community, providing insights into quality, common challenges, emerging technologies, and evolving developer needs. Silva et al. [11] report that ChatGPT has significantly impacted SO, offering fast, human-like responses that have raised questions about the platform's future in the AI era. The study noted a decline in overall SO activity, though some communities remain active. Both models excel at addressing general programming queries but struggle with specific frameworks and libraries, leading developers to return to SO when LLMs fall short. Similarly, Zhong et al. [6] developed the RobustAPI dataset, featuring 1,208 coding questions from SO related to 18 Java APIs. Their study revealed that even advanced models like GPT-4 produced API misuses in 62% of the generated code, posing risks when applied to real-world software development.

Nonetheless, there is no study that investigates the quality of LLaMA and its adoption in practice. This study fills in the gap by studying the discussions related to LLaMA on SO discussions.

3. Methodology

As shown in Figure 1, a motivating example is a Stack Overflow post where a user inquires about installing the `LLaMA-cpp-python` package. This post has garnered 38,975 views (at the time of writing), illustrating the widespread interest in LLaMA but also highlighting that developers frequently encounter challenges requiring external help. Despite its growing popularity, the installation and configuration of LLaMA packages remain common stumbling blocks.

In light of this, our research focuses on examining the discussions surrounding LLaMA on Stack Overflow. By analyzing these interactions, we aim to uncover the most prevalent issues and limitations faced by developers working with LLaMA. This study not only seeks to identify key challenges but also offers valuable insights for both novice users looking to get started with LLaMA and experienced developers seeking to optimize and enhance their implementations. Ultimately, our findings will contribute to improving the support and resources available to the LLaMA

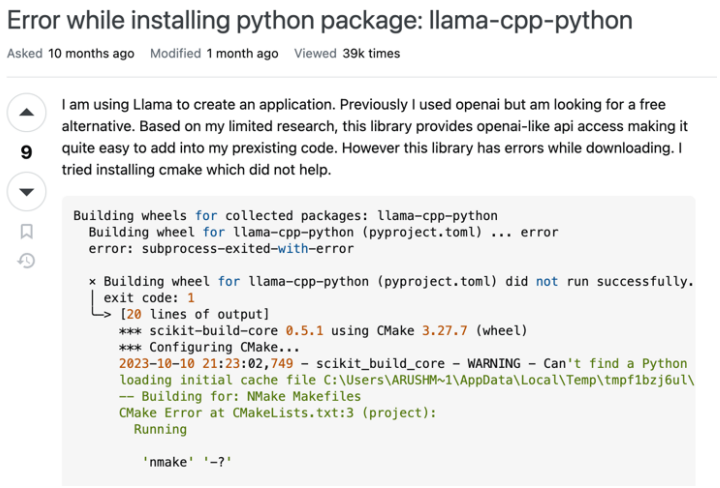

Figure 1: A LLaMA question of Stack Overflow (Post ID 77267346)

community, facilitating smoother adoption and integration of the model into various workflows.

We ask the following research questions in this study.

1. *RQ1: What are the topics of discussion about LLaMA on Stack Overflow?* We desire to identify and categorize the topics of discussion related to "LLaMA" on Stack Overflow. This is to determine the most common themes and issues raised by the developer community concerning LLaMA.
2. *RQ2: What are the related topics when discussing LLaMA on Stack Overflow?* The second research question focused on identifying related tags co-occurring with the LLaMA tag on Stack Overflow. This is to find other relevant topics or challenges that LLaMA users may face or need to study.

This section details the steps undertaken to address the two research questions posed earlier. As illustrated in Figure 2, our methodology involves three key phases: data collection, preprocessing, and analysis. Each phase is designed to ensure a systematic and thorough examination of Stack Overflow discussions related to LLaMA. In the data collection phase, we gathered relevant posts from Stack Overflow, ensuring a representative sample of developer interactions. This was followed by the preprocessing phase, where we cleansed and refined the data to ensure its quality and relevance for analysis. Finally, the analysis phase involved categorizing the posts and performing keyword frequency analysis to uncover common themes and patterns.

3.1. Data Collection

Our study is based on data collected directly from Stack Overflow, particularly focusing on posts related to LLaMA, the generative AI model from Meta. Initially, we considered using the Stack Overflow public data dump files, including `Posts.xml` and `Tags.xml`[3]. However, after downloading and inspecting these files, we found that they did not contain recent posts relevant to our study, particularly those involving technologies like LLaMA, likely due to the release of LLaMA being more recent than the last update of the data dump.

[3]https://archive.org/details/stackexchange

As a result, we adopted a more direct and up-to-date data collection approach. We utilized the web scraping tool[4] to scrape posts directly from Stack Overflow. The scraping process was conducted on July 22, 2024. To comply with Stack Overflow's usage policies and avoid overloading their servers, we incorporated waiting times between requests. The data collected included the posts' links, titles, bodies, and tags.

To effectively capture posts related to "LLaMA", we employed two distinct methods:

Method 1: Keyword Search — We conducted a search on Stack Overflow using the keyword "LLaMA"[5]. This search yielded **2,405 posts**, which we categorized as follows:

- **Title Group (644 posts)**: Posts where "LLaMA" appeared in the title.
- **Body Group (1,761 posts)**: Posts where "LLaMA" appeared in the body. However, after manual inspection, many of these posts were deemed irrelevant and thus excluded from further analysis.

Method 2: Tag Search (770 posts) — We also searched for posts tagged with "LLaMA" on Stack Overflow[6]. This search resulted in 770 posts, which were compiled into a separate group called the **Tag Group**.

3.2. Data Preprocessing

Data preprocessing was essential to ensure the relevance and quality of the data used in our analysis. The following steps were undertaken to refine the data:

Step 1: Tag Separation — The tags in the Tag Group were initially compiled as a single string. To analyze the tags associated with each post more precisely, we separated them into individual tags, enabling more effective identification and analysis.

Step 2: Duplicate Removal — During preprocessing, we identified overlaps between the Title Group and Tag Group, as some posts appeared in both groups due to being tagged

[4]Web Scraper version 1.87.6 (available at: https://webscraper.io/)
[5]We queried from the URL https://stackoverflow.com/search?tab=newest&q=LLaMA&searchOn=3
[6]We queried from the URL https://stackoverflow.com/questions/tagged/LLaMA?tab=Newest

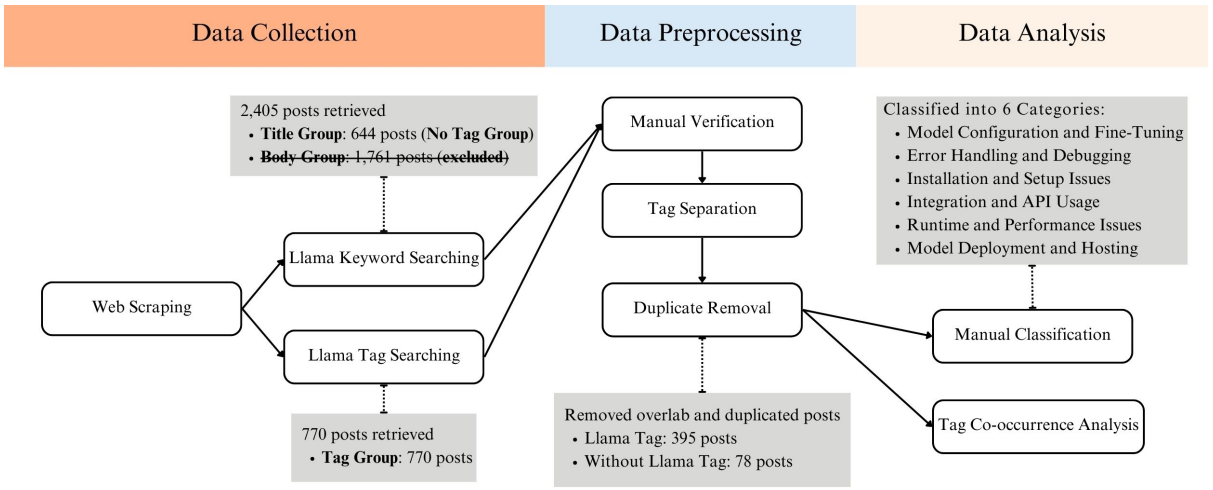

Figure 2: The Experimental Procedure

with "LLaMA." Additionally, we detected duplicate entries with identical post links and titles. These redundancies were removed, resulting in a refined dataset of 473 posts comprising 395 posts tagged with "LLaMA" and 78 posts without the tag.

3.3. Dataset Characteristics

After data collection and preprocessing, our final dataset consisted of 473 posts, all centered on LLaMA-related topics. These posts cover a range of issues, questions, and discussions about LLaMA, including configuration, usage, and challenges.

For instance, a typical post in our dataset may include a query about fine-tuning the LLaMA model:

> *"How do I fine-tune the LLaMA model on a custom dataset? I'm facing memory issues during training and could use some advice on optimizing performance."*

Another example might address integration issues:

> *"I'm trying to integrate LLaMA with an existing API but keep encountering errors during the authentication process. Has anyone faced similar issues?"*

These examples illustrate the types of discussions that form the basis of our subsequent analysis.

3.4. Data Analysis

Using the cleansed datasets, we analyzed the topics of discussion related to LLaMA on Stack Overflow to address our research questions:

RQ1: What are the common topics discussed regarding LLaMA? — We manually classified the titles and bodies of the posts to identify common topics. To ensure thoroughness, the first author initially skimmed through all posts to get a sense of the themes and formulated the six categories as a preliminary structure. Then the first and second authors independently reviewed all posts, categorizing them into six groups: *Model Configuration and Fine-Tuning, Error Handling and Debugging, Installation and Setup Issues, Integration and API Usage, Runtime and Performance Issues,* and *Model Deployment and Hosting*. These six groups were established before the manual classification by the first authors during the data collection and data preprocessing steps. One post could fall into multiple categories. Any disagreements were resolved through discussion until a consensus was reached.

RQ2: What related topics and technologies are associated with LLaMA? — We examined the tags associated with "LLaMA" to identify related topics and technologies. The co-occurrence of these tags with "LLaMA" shows the broader technological ecosystem and application areas linked to LLaMA.

4. Results

This section presents the findings from our analysis of the discussions related to the LLaMA model on Stack Overflow and the answers to our research questions. We address the research questions (RQ1 and RQ2) through a detailed examination of the collected and cleansed datasets.

4.1. Answering RQ1

To answer RQ1, we manually categorized the posts into six distinct categories based on the nature of the issues discussed. To assess the reliability of the manual classification, we calculated the inter-rater reliability using Cohen's Kappa statistic. The Kappa score was 0.883, indicating an almost perfect agreement between the two authors. The categorization helped us to identify the most common themes in the developer community's conversations about LLaMA. Table 1 provides a summary of the categories and the number of posts that relate to each category.

From our analysis, it is evident that the majority of discussions focus on *Model Configuration and Fine-Tuning*, with 135 posts, making it the most frequently discussed topic. This suggests that many developers are struggling with configuring and fine-tuning LLaMA models to meet specific needs. Posts in this category often mention challenges such as adjusting hyperparameters, loading pre-trained models,

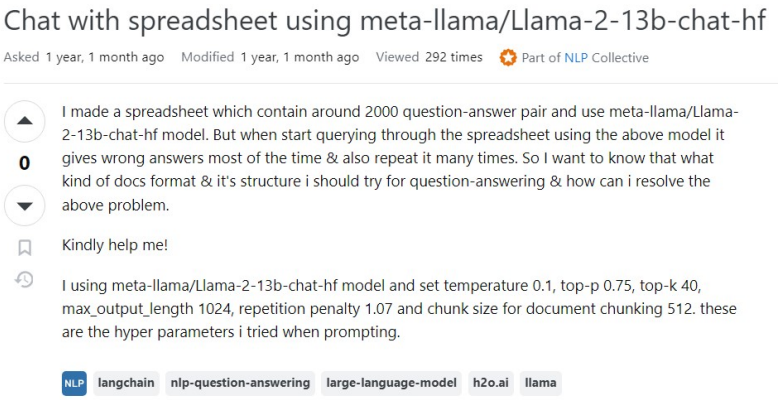

Figure 3: Example of Model Configuration and Fine-Tuning post (Post ID 76880690)

Table 1
Categories of LLaMA Discussion on Stack Overflow

Category	Number of Posts
Model Configuration and Fine-Tuning	135
Error Handling and Debugging	110
Installation and Setup Issues	91
Integration and API Usage	86
Runtime and Performance Issues	73
Model Deployment and Hosting	24
Total	**519**

and optimizing models for specific tasks or datasets. The prevalence of this category suggests that LLaMA's flexibility and complexity in configuration require careful attention and often lead to challenges that developers seek to overcome. Posts in this category commonly address issues like adjusting hyperparameters, loading pre-trained models, and optimizing models for particular tasks or datasets. The prominence of this category indicates that LLaMA's flexibility and complexity in configuration often present challenges that developers actively seek to resolve. Figure 3 shows a Stack Overflow post titled "Chat with spreadsheet using Meta Llama (Llama 2 13B Chat HF)," categorized under the Model Configuration and Fine-Tuning category. In this post, the questioner is facing the problem of using LLaMA for querying spreadsheet data.

Error Handling and Debugging, accounting for 110 posts. This category includes posts where developers encountered errors during the use of LLaMA and sought solutions to resolve these issues. Common topics in this category involve troubleshooting runtime errors, resolving compatibility issues with other libraries, and debugging scripts that fail to execute as expected. The prevalence of posts in this category underscores the need for robust debugging tools and clear documentation to help developers efficiently resolve issues. Figure 4 depicts a Stack Overflow post titled "How to debug the Llama 2 inference command with VSCode," which is categorized under "Error Handling and Debugging." In this post, the questioner asks about configuring Visual Studio Code to debug the Llama 2 inference script.

Installation and Setup Issues is another prominent category, comprising 91 posts. This category covers problems encountered during the initial stages of working with LLaMA, including installation errors, environment configuration challenges, and difficulties in setting up dependencies. The high number of posts in this category indicates that getting started with LLaMA can be particularly challenging, especially for users who are new to the model or unfamiliar with the broader ecosystem of tools it integrates with. Figure 5 shows a Stack Overflow post titled "Cuda 12.2 and issue with bitsandbytes package installation" categorized under "Installation and Setup Issues." In this post, the developer is facing an issue with running Llama 2 on Google Colab and asks for help.

Integration and API Usage, with 86 posts, reflects discussions on how to connect LLaMA with other systems, particularly through APIs. Developers often seek guidance on integrating LLaMA into existing workflows, leveraging its capabilities alongside other tools, and addressing API-related challenges. These discussions highlight the importance of seamless integration between LLaMA and other technologies, as well as the need for clear guidelines on API usage.

Runtime and Performance Issues, comprising 73 posts, focuses on challenges that developers face during the execution of LLaMA models. This includes discussions on optimizing model performance, managing resource consumption, and addressing latency issues. Posts in this category often highlight the need for efficient execution of LLaMA models, especially in production environments where performance is critical.

Model Deployment and Hosting, with 24 posts, is the least discussed category. Posts here focus on deploying LLaMA models into production, managing model versions, and hosting models on different platforms. The relatively low number of posts in this category might suggest that deployment is a more advanced stage of working with LLaMA, which fewer users have reached, or that deployment-related issues are less frequent or already well-documented within the community.

Overall, the distribution of posts across these categories provides valuable insights into the areas where LLaMA users are most likely to encounter challenges. It also highlights the importance of comprehensive support and resources in the areas of model configuration, error handling, and integration.

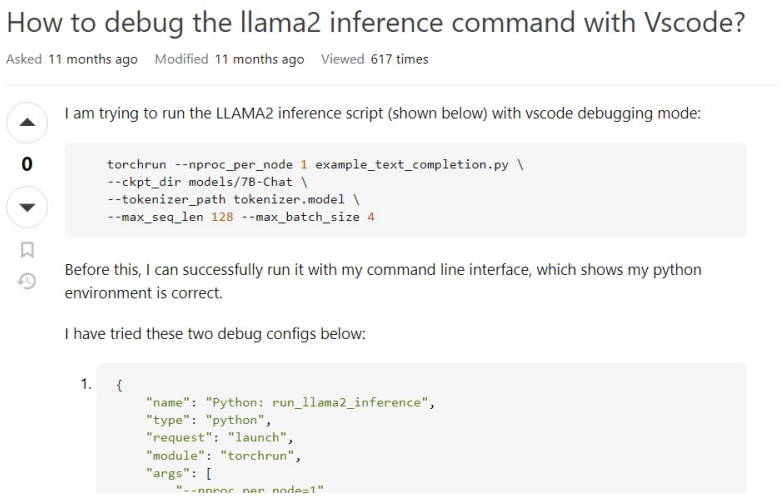

Figure 4: Example of the Error Handling and Debugging post (Post ID 77421713)

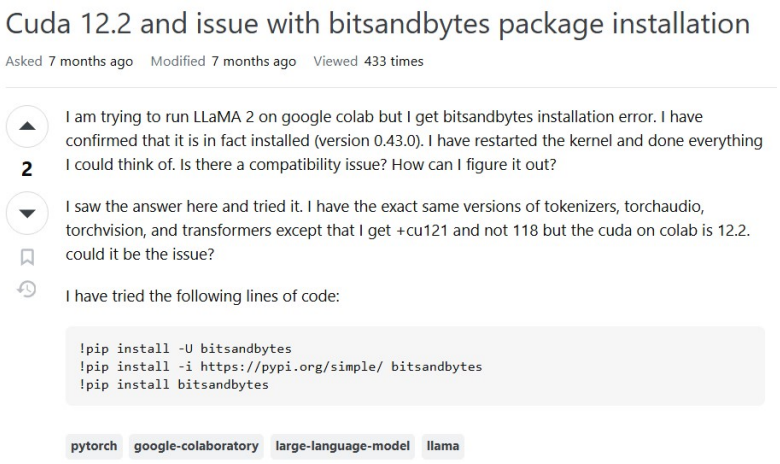

Figure 5: Example of Installation and Setup Issues post (Post ID 78194505)

4.2. Answering RQ2

To address RQ2, we examined the co-occurrence of tags in posts discussing LLaMA. By analyzing these tags, we aimed to identify related topics and technologies that are commonly mentioned alongside LLaMA on Stack Overflow. Table 2 summarizes the frequency of the most common co-occurring tags.

The analysis revealed that the *large-language-model* tag was the most frequently co-occurring tag with *LLaMA*, appearing in 201 posts. This suggests that discussions around LLaMA are often framed within the broader context of large language models, indicating that developers are considering LLaMA alongside other major models in this category. The frequent mention of *python* (184 posts) and *huggingface-transformers* (109 posts) indicates that developers are actively using Python-based tools and libraries, particularly Hugging Face's Transformers library, to work with LLaMA. This reflects LLaMA's integration into the Python ecosystem and its compatibility with popular machine-learning frameworks.

The co-occurrence of tags like *langchain* (77 posts) and *pytorch* (70 posts) further supports the observation that LLaMA is frequently used in conjunction with other machine-learning tools. LangChain, in particular, is a framework designed for building applications with LLMs, suggesting that LLaMA users are developing complex workflows that involve multiple LLMs.

Notably, the *openai-api* tag appeared in 26 posts, indicating a significant interest in interoperability between LLaMA and OpenAI's models. The posts in this category reveal several common themes:

1. *Interoperability Between LLaMA and OpenAI Models:* Many posts discuss how to integrate or migrate between LLaMA models and OpenAI APIs. For instance, questions related to migrating from ChatGPT to Llama 2 or using different LlamaIndex chat engine modes with an OpenAI key suggest that users are exploring how to use both systems together or comparing their functionalities.
2. *LangChain and LLaMA:* Several posts mention LangChain in conjunction with LLaMA. LangChain is a framework for building applications with LLMs, and the discussions around using it with LLaMA suggest that users are working on sophisticated workflows involving multiple language models. This highlights LLaMA's role in the broader landscape of lan-

Table 2
Top Co-Occurring Tags with llama Tag on Stack Overflow

Tag	Occurrences
llama	398
large-language-model	201
python	184
huggingface-transformers	109
langchain	77
pytorch	70
huggingface	66
llama-index	40
nlp	39
artificial-intelligence	34
fine-tuning	28
openai-api	26
machine-learning	25
ollama	23
llamacpp	22
llama-cpp-python	22
llama3	20
python-3.x	20
amazon-sagemaker	19
chatbot	14
gpu	14
amazon-web-services	12
Others	356
Total	**1,819**

guage model applications.

3. *Fine-Tuning and Model Performance:* With the *fine-tuning* tag appearing in 28 posts, this category reflects discussions around optimizing LLaMA models. Posts such as *"LLaMA Index training my own model gives poor results"* indicate challenges in fine-tuning LLaMA models. Users seek advice on improving model performance, particularly in fine-tuning and optimizing models for specific tasks.
4. *Vector Databases and RetrievalQA:* Discussions in this area involve using LLaMA models with vector databases and RetrievalQA. Users are focusing on effectively retrieving documents or managing storage when integrating LLaMA with OpenAI's API, reflecting the complexity of tasks users are undertaking.
5. *Computational Resources:* Questions related to hardware usage, such as issues with running LLaMA models on CPUs or optimizing GPU usage, highlight developers' concerns about the computational demands of LLaMA models. Tags such as *gpu* and *amazon-sagemaker* appear alongside discussions focused on resource optimization.

These findings illustrate that LLaMA is part of a larger ecosystem of tools and technologies, with significant interest in how it can be integrated with or compared to other models, particularly those from OpenAI. The discussions also underscore the importance of effective model management, performance optimization, and resource utilization when working with LLaMA.

4.3. Threats to Validity

Several threats to validity may impact the findings of this study. *Internal Validity:* One potential threat is the assumption that all posts in the dataset were relevant to Meta's LLaMA AI. This assumption may have resulted in the inclusion of irrelevant or off-topic content. We mitigated this risk by performing a manual verification of 500 posts to ensure relevance, though some less obvious irrelevant content might still remain. Additionally, our reliance on manual classification introduces the risk of human error and bias. To address this, two authors independently classified the posts, and any discrepancies were resolved through discussion to increase consistency and reduce subjectivity. However, biases inherent in manual processes may still exist, and the absence of automated classification tools may have limited the scalability of the analysis.

Furthermore, the data collection was conducted only up until July 22, 2024, which excludes newer posts. As the field of large language models (LLMs) evolves rapidly, this limitation may have prevented us from capturing recent trends or emerging challenges, potentially affecting the completeness and timeliness of our analysis. *External Validity:* The findings are based solely on Stack Overflow (SO) posts with the keyword "LLaMA" in the titles or tags, which may limit the generalizability of our results to other technical Q&A platforms such as GitHub, Reddit, or specialized forums where different types of discussions and more complex technical issues may be addressed. By focusing exclusively on SO, we may have missed richer, more nuanced developer challenges that could provide a broader understanding of LLaMA adoption across different communities.

5. Implications

The findings from this study provide valuable insights into the quality of Meta's LLaMA model and how the developer community engages with it on Stack Overflow, particularly in terms of overcoming technical challenges. The analysis reveals that discussions predominantly focus on issues such as configuring, fine-tuning, and integrating LLaMA into various applications. This highlights the model's flexibility but also points to its complexity, underscoring the need for improved documentation, resources, and tools.

One key implication is the necessity for enhanced community support and resources for model configuration and fine-tuning. The frequency of posts on these topics suggests that many developers, especially those without advanced expertise in machine learning, encounter significant difficulties. By improving documentation and offering more user-friendly tools, Meta could lower the barrier to entry for a wider audience, leading to broader adoption of LLaMA. This could also include the development of community-driven forums, FAQs, or official support channels dedicated to troubleshooting configuration and fine-tuning issues.

Another important implication is the need to prioritize seamless integration with existing machine-learning ecosystems. The co-occurrence analysis shows that LLaMA is frequently used in conjunction with popular frameworks like Hugging Face's Transformers, PyTorch, and LangChain, particularly in Python environments. This suggests that future iterations of LLaMA should focus on making integration with these frameworks more straightforward and efficient, potentially through more robust APIs, pre-built connectors, or better interoperability guidelines. Ensuring compatibility with widely-used tools will be crucial in positioning LLaMA as a go-to solution for developers working on real-world applications. Finally, the relatively low number of posts discussing the deployment and hosting of LLaMA models

suggests that this is still an emerging area. However, as more developers move toward deploying LLaMA models in production environments, there will likely be an increasing demand for comprehensive deployment tools, best practices, and infrastructure support.

6. Conclusion and Future Work

In conclusion, this preliminary study provides a detailed investigation of the quality, the challenges, and related topics in the Stack Overflow community's discussions about LLaMA. By understanding these areas, Meta and the broader developer community can better support the use of LLaMA, ultimately driving innovation in LLM development.

This study provides valuable insights into the challenges developers face when adopting LLaMA, based on Stack Overflow discussions. However, several areas for future research could significantly enrich the findings and address the limitations identified in this study. First, expanding the dataset to include posts beyond July 2024 will help capture evolving trends as LLaMA and other large language models (LLMs) continue to develop. Additionally, incorporating data from other platforms such as GitHub Issues, Reddit, and developer forums could provide a broader perspective on LLaMA's usage, especially on more complex technical problems and nuanced discussions that may not be captured on Stack Overflow alone. Comparing LLaMA to other LLMs, such as ChatGPT or Claude, would also provide valuable insights, allowing researchers to understand LLaMA's challenges in the broader landscape and better justify its focus.

Furthermore, future research should enhance the methodology by employing a more rigorous approach to data filtering and analysis. Pre-processing the data to exclude trivial questions and focusing on more substantial challenges would yield more meaningful insights. Using established qualitative coding frameworks for topic classification would further improve the transparency and validity of the analysis. Another promising direction is incorporating sentiment analysis to understand community attitudes toward LLaMA. By analyzing the tone of discussions across platforms, researchers could uncover whether developers' experiences with LLaMA are generally positive, negative, or neutral, offering Meta and the developer community actionable feedback for improving the tool.

Additionally, complementing the analysis with user studies—such as surveys or interviews—could provide a deeper understanding of the practical challenges faced by developers using LLaMA in real-world scenarios. Exploring specific use cases where LLaMA is integrated into different application domains, such as natural language processing (NLP) or enterprise applications, could reveal unique challenges and benefits in various contexts. Finally, investigating advanced stages of LLaMA adoption, particularly in production environments, would help identify issues related to deployment and model hosting, offering a more complete picture of LLaMA's practical applications and limitations. By addressing these areas, future research will contribute to a more comprehensive understanding of LLaMA's role within the LLM ecosystem, driving more effective support for developers and fostering broader adoption of open-source LLMs.

7. ACKNOWLEDGEMENT

This research project was supported by the Faculty of ICT, Mahidol University.

References

[1] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lacha ux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al., Llama: Open and efficient foundation language models, arXiv preprint arXiv:2302.13971 (2023).

[2] S. Sadeghi, A. Bui, A. Forooghi, J. Lu, A. Ngom, Can large language models understand molecules?, 2024. URL: https://arxiv.org/abs/2402.00024. arXiv:2402.00024.

[3] K. Wangsa, S. Karim, E. Gide, M. Elkhodr, A systematic review and comprehensive analysis of pioneering ai chatbot models from education to healthcare: Chatgpt, bard, llama, ernie and grok, Future Internet 16 (2024). URL: https://www.mdpi.com/1999-5903/16/7/219. doi:10.3390/fi16070219.

[4] J. Son, B. Kim, Trend Analysis of Large Language Models through a Developer Community: A Focus on Stack Overflow, Information 14 (2023).

[5] A. Hörnemalm, O. Norberg, T. Mejtoft, ChatGPT as a Software Development Tool The Future of Development, Master's thesis, Umeå University, Department of Applied Physics and Electronics, 2023.

[6] L. Zhong, Z. Wang, Can llm replace stack overflow? a study on robustness and reliability of large language model code generation, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 38, 2024, pp. 21841–21849.

[7] K. Jin, C.-Y. Wang, H. V. Pham, H. Hemmati, Can ChatGPT Support Developers? An Empirical Evaluation of Large Language Models for Code Generation, in: Proceedings of the 21st International Conference on Mining Software Repositories, MSR '24, 2024, p. 167–171.

[8] A. Fan, B. Gokkaya, M. Harman, M. Lyubarskiy, S. Sengupta, S. Yoo, J. M. Zhang, Large Language Models for Software Engineering: Survey and Open Problems, in: ICSE-FoSE'23, 2023, pp. 31–53.

[9] A. Buscemi, D. Proverbio, Chatgpt vs gemini vs llama on multilingual sentiment analysis, 2024. URL: https://arxiv.org/abs/2402.01715. arXiv:2402.01715.

[10] A. Salinas, P. Shah, Y. Huang, R. McCormack, F. Morstatter, The unequal opportunities of large language models: Examining demographic biases in job recommendations by chatgpt and llama, in: Proceedings of the 3rd ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization, EAAMO '23, Association for Computing Machinery, New York, NY, USA, 2023. URL: https://doi.org/10.1145/3617694.3623257. doi:10.1145/3617694.3623257.

[11] L. Da Silva, J. Samhi, F. Khomh, Chatgpt vs llama: Impact, reliability, and challenges in stack overflow discussions, arXiv preprint arXiv:2402.08801 (2024).