Enhancing Fault Isolation in Hardware Systems Using Large Language Models (LLM)

Miglioramento dell'isolamento dei guasti nei sistemi hardware utilizzando modelli linguistici di grandi dimensioni (LLM)

Supervisor: Prof. Gianfranco Lombardo

Student: Matteo Gianvenuti, student ID: 364979

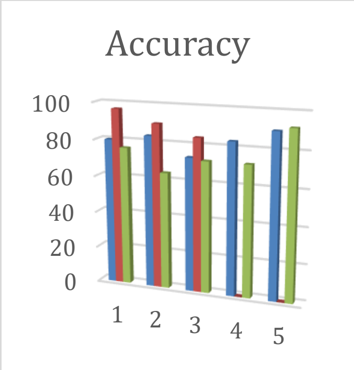
Abstract:

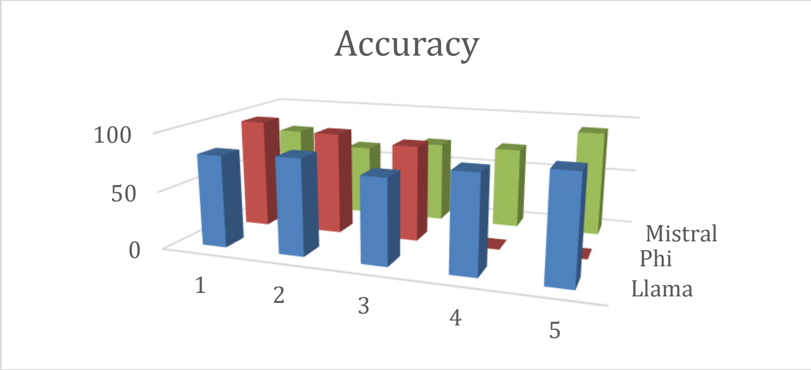
The hardware systems evolution led to an increasing complexity of the diagnostics processes and architectures. In industrial and mission-critical environments, like the defence, the ability of quickly identify and isolate faults is fundamental to guarantee reliability, security and availability. However, the traditional methods of fault isolation are usually and typically based on static rules, expert systems, manual analysis and instruments limited in unstructured data analysis. In parallel, Large Language Models (LLMs) have demonstrated a surprisingly capacities in comprehension of the natural language and reasoning tasks. Especially in the case in which models know the environment in which they operate.

In this work thesis I want to explore the use of LLMs in hardware diagnostic to analyse diagnostics test results, identify any possible anomalies and try to find their possible causes. I want to reach these objectives in an offline working environment characterized by challenging limited resources, like the resources available on computer laptop. In particular, this work thesis explores the efficacy of a diagnostic framework based on a LLM executed locally without the support of cloud infrastructures or high-end GPU. This approach results particularly interesting in industrial scenarios in which the connection is absent, not guaranteed or the access to advanced hardware resources is constrained by hardware or budget requirements. The choice of operating in a restricted hardware environment does not represent a limitation but rather a strategic experimental condition to evaluate the scalability and adaptability of LLMs in real and decentralized contexts. The local execution of LLMs guarantee privacy and control of the test environment.

This work is based on In-Context Learning and Prompt Engineering to reach the cited objectives with limited resources. To reach that objectives I structured the work thesis in two steps. A first, consisting in verify if the LLMs, appropriately learned with In-Context Learning techniques digest and understand the structure and the functioning of a complex hardware system. A second, consisting in evaluating the LLMs ability of analyse logs and telemetry data generated by the diagnostic software of the company to identify anomalies and their causes. To carry out the work activities of the second step it was necessary to create a simulation environment with various scenarios and a specific dataset for each scenario. I also implemented the Open WebUI graphical interface to provide a better experience and usage of the LLM based solution.

To reach the objectives I experimented especially with three models: Llama3.2-3B-Instruct, Phi-4-mini-reasoning and Mistral-7B-Instruct-V0.3. Follows a chart that compares the models' accuracy for activities of the first step. It shows that Llama has the best average performance. For that reason, I decided to keep Llama as main LLM for the second step activities. Llama also demonstrated to have the best performance in natural language comprehension, generalizing capabilities and the lowest response time.





The few-shot examples should help the model in understanding its tasks and lead to better results. This is because the model has more knowledge on how it should analyse the diagnostics tests. However, if there is a great amount of data, in this case tests and their results, creates a large context input for a small model, making difficult for it a “complete understanding” of the situation and to focus on the real issue. In fact, I experimented with bigger commercial models like ChatGPT the model task, these enormous models can understand the situation and complete the task more accurately. In this case the size of the model leads the few-shot technique to worse performance than the cases with a general prompt with the inference logics only due to the huge amount of data for a small model.

Also, the number of tests has a great impact on the performance of this approach. In fact, with less data the model was able to better understand the situation and identify anomalies.

The applied solution demonstrated also the importance of the representation. In fact, with Json for the test representation, the model was able to “understand” more. As it demonstrated that the performance widely depends on the model/input size.

In conclusion, this approach demonstrated that the LLMs could identify most of the errors making the diagnostics faster, but sometimes small LLMs have difficulties in reasoning and identifying the possible causes of these errors with many data due to their limited memorization and reasoning capacities.

As future development this work could be integrated into the diagnostics software of the company. For example, with a dedicated button could be possible to take the tests data for a use case and feed them to the model to receive a recap/analysis answer. Furthermore, with more data could be possible to perform a fine-tune rather than In-Context Learning. A fine-tune could drastically improve the performance of this approach.