



FEDERAL UNIVERSITY OF SANTA CATARINA
TECHNOLOGY CENTER
AUTOMATION AND SYSTEMS DEPARTMENT
UNDERGRADUATE COURSE IN CONTROL AND AUTOMATION ENGINEERING

Douglas Mateus Machado

**Development of a Cognitive Assistant for Industrial Maintenance Based on
Retrieval-Augmented Generation (RAG) at STMicroelectronics**

Grenoble, France

2024

This document is the property of STMicroelectronics; it may not be communicated to third parties and/or reproduced without the prior written permission of STMicroelectronics, and its content may not be disclosed.

Douglas Mateus Machado

**Development of a Cognitive Assistant for Industrial Maintenance Based on
Retrieval-Augmented Generation (RAG) at STMicroelectronics**

Final report of the subject DAS5511 (Course Final Project) as a Concluding Dissertation of the Undergraduate Course in Control and Automation Engineering of the Federal University of Santa Catarina.
Supervisor: Prof. Eric Aislan Antonelo, Dr.
Co-supervisor: Philippe Vialletelle, Eng.

Grenoble, France
2024

Ficha catalográfica gerada por meio de sistema automatizado gerenciado pela BU/UFSC.
Dados inseridos pelo próprio autor.

Mateus Machado, Douglas
Development of a Cognitive Assistant for Industrial
Maintenance Based on Retrieval-Augmented Generation (RAG)
at STMicroelectronics / Douglas Mateus Machado ;
orientador, Eric Aislan Antonelo, coorientador, Phillipe
Vialletelle, 2024.
66 p.

Trabalho de Conclusão de Curso (graduação) -
Universidade Federal de Santa Catarina, Centro Tecnológico,
Graduação em Engenharia de Controle e Automação,
Florianópolis, 2024.

Inclui referências.

1. Engenharia de Controle e Automação. 2. Retrieval
Augmented Generation. 3. Industrial Maintenance. 4.
Generative Artificial Intelligence. I. Aislan Antonelo,
Eric. II. Vialletelle, Phillipe. III. Universidade Federal
de Santa Catarina. Graduação em Engenharia de Controle e
Automação. IV. Título.

Douglas Mateus Machado

Development of a Cognitive Assistant for Industrial Maintenance Based on Retrieval-Augmented Generation (RAG) at STMicroelectronics

This dissertation was evaluated in the context of the subject DAS5511 (Course Final Project) and approved in its final form by the Undergraduate Course in Control and Automation Engineering

Grenoble, July 10, 2024.

Prof. Marcelo De Lellis Costa de Oliveira, Dr.
Course Coordinator

Examining Board:

Prof. Eric Aislan Antonelo, Dr.
Advisor
UFSC/CTC/DAS



Documento assinado digitalmente
Eric Aislan Antonelo
Data: 15/07/2024 11:06:36-0300
CPF: ***.163.409-**
Verifique as assinaturas em <https://v.ufsc.br>

Philippe Vialletelle, Eng.
Supervisor
STMicroelectronics

Thiago Fonseca Rech, Eng.
Evaluator
UFSC/CTC/INEP

Prof. Eduardo Camponogara, Dr.
Board President
UFSC/CTC/DAS

I dedicate this work to my children, Bernardo and Cicero, who motivate me to be a better man, and to my beloved wife Alexandra who has blessed me with a beautiful family and supports me unconditionally. To my sister Samantha and to my parents, José and Carmen, who have always supported me and helped me reach further than I ever imagined

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my former internship supervisors, Zakaria Yahouni and Gulgun Alpan, for their support and recommendation to carry out this work.

I also thank my internship tutor, Philippe Vialletelle, for giving me the opportunity to undertake this internship at a world-renowned company like STMicroelectronics.

I would like to express my gratitude to Pascal Urard, Roberto Guizzetti and Patricia Rollandet, for the technical and managerial guidance provided to me during the internship. Their experience and knowledge were crucial for the development of this work.

I cannot fail to mention my colleagues Maelig Hurte, Awa Seck, Soumaya Metrab, Arthur Fournier and Baptiste Carrolaggi who shared the project's challenges and the knowledge gained during the period. Their collaboration was essential for the accomplishment of this work.

To Lucas Andrade Militão, for guiding me in my research journey, teaching me about the significance of science, and the impact of diligence and persistence.

I would like to thank the Data Scientist from the Manufacturing Data Science Team, Michel Juge, who supported me in the data extraction and acquisition for the development of the project. Also João Paulo Brum for positive exchanges.

Thanks to my colleague and friend Michel Chevalier for the support during my internship.

Special thanks to Marc Grosso and Benoit Manini from the Rousset maintenance team who, as users and supporters of the project, provided relevant ideas, positive feedback, encouragement, and support during the project.

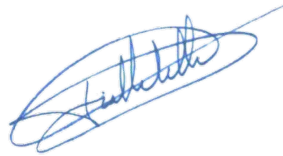
Lastly, I would like to thank the company that welcomed me, STMicroelectronics, and the BooSTcamp program, which provided a great work environment for my technical and professional development. I am grateful for the opportunity to work on a challenging and innovative project, and for the trust placed in me.

"It is change, continuous change, inevitable change, that is the dominant factor in society today. No sensible decision can be made anymore without taking into account not only the world as it is, but the world as it will be."
ASIMOV, 1978

DISCLAIMER

Grenoble, July 10th, 2024.

As representative of the STMicroelectronics in which the present work was carried out, I declare this document to be exempt from any confidential or sensitive content regarding intellectual property, that may keep it from being published by the Federal University of Santa Catarina (UFSC) to the general public, including its online availability in the Institutional Repository of the University Library (BU). Furthermore, I attest knowledge of the obligation by the author, as a student of UFSC, to deposit this document in the said Institutional Repository, for being it a Final Program Dissertation ("*Trabalho de Conclusão de Curso*"), in accordance with the *Resolução Normativa n° 126/2019/CUn*.



Philippe Vialletelle
STMicroelectronics

ABSTRACT

This work presents the development of a cognitive assistant for industrial maintenance at STMicroelectronics, using the *Retrieval-Augmented Generation* (RAG) methodology. Industrial maintenance is crucial to ensure the availability of machines, the safety of operators, and the sustainability of production processes. Traditionally, maintenance strategies such as corrective, preventive, and predictive are employed to optimize machine operation. However, the emergence of Industry 4.0 and 5.0 demands more advanced solutions, such as *Prescriptive Maintenance* (RxM), which integrates artificial intelligence to improve decision-making. In this context, the developed cognitive assistant employs the RAG technique, which combines generative language models, such as *Generative pre-trained transformer* (GPT), with the retrieval of specific information to enrich the system's response capability. This method allows the assistant to access technical documents and maintenance records from STMicroelectronics, expanding its knowledge beyond the data with which it was originally trained. The RAG approach helps to mitigate problems of hallucinations and lack of interpretability common in purely generative models, providing more accurate and contextualized responses. The project demonstrates how the combination of generative artificial intelligence with specific information retrieval can create a powerful tool to support industrial maintenance, aligning with the Industry 5.0 principles of putting the human at the center of technological interactions. The evaluation method categorized responses into five levels: completely false (weight -5), partially false (weight -3), factually correct but incomplete (weight 1), fully correct (weight 3), and fully correct with useful reasoning (weight 5). This approach ensured robust assessment by penalizing hallucinations and rewarding high-quality responses. Results from four qualitative evaluation sessions showed the cognitive assistant, tested with 45 question-answer pairs, achieved an average score of 3.27, a normalized average of 82.67 indicating complete, correct and useful information on the majority of tested cases, and an average response time of 32.50 seconds, demonstrating its effectiveness in supporting industrial maintenance tasks.

Keywords: Retrieval-Augmented Generation. Industrial Maintenance. Generative Artificial Intelligence.

RESUMO

Este trabalho apresenta o desenvolvimento de um assistente cognitivo para manutenção industrial na STMicroelectronics, utilizando a metodologia RAG. A manutenção industrial é crucial para garantir a disponibilidade das máquinas, a segurança dos operadores e a sustentabilidade dos processos de produção. Tradicionalmente, estratégias de manutenção como corretiva, preventiva e preditiva são empregadas para otimizar a operação das máquinas. No entanto, o surgimento da Indústria 4.0 e 5.0 exige soluções mais avançadas, como RxM, que integra inteligência artificial para melhorar a tomada de decisões. Nesse contexto, o assistente cognitivo desenvolvido emprega a técnica RAG, que combina modelos de linguagem generativa, como GPT, com a recuperação de informações específicas para enriquecer a capacidade de resposta do sistema. Este método permite que o assistente acesse documentos técnicos e registros de manutenção da STMicroelectronics, expandindo seu conhecimento além dos dados com os quais foi originalmente treinado. A abordagem RAG ajuda a mitigar problemas de alucinações e falta de interpretabilidade comuns em modelos puramente generativos, fornecendo respostas mais precisas e contextualizadas. O projeto demonstra como a combinação de inteligência artificial generativa com a recuperação de informações específicas pode criar uma ferramenta poderosa para apoiar a manutenção industrial, alinhando-se aos princípios da Indústria 5.0 de colocar o ser humano no centro das interações tecnológicas. O método de avaliação categorizou as respostas em cinco níveis: completamente falso (peso -5), parcialmente falso (peso -3), factualmente correto, mas incompleto (peso 1), totalmente correto (peso 3) e totalmente correto com raciocínio útil (peso 5). Esta abordagem garantiu uma avaliação robusta, penalizando alucinações e recompensando respostas de alta qualidade. Os resultados de quatro sessões de avaliação qualitativa mostraram que o assistente cognitivo, testado com 45 pares de perguntas e respostas, obteve uma pontuação média de 3,27, uma média normalizada de 82,67, indicando informações completas, corretas e úteis na maioria dos casos testados, e um tempo de resposta médio de 32,50 segundos, demonstrando sua eficácia no suporte às tarefas de manutenção industrial.

Palavras-chave: Retrieval-Augmented Generation. Manutenção Industrial. Generative Artificial Intelligence.

LIST OF FIGURES

Figure 1 – Framework’s solution for the augmented maintenance.	21
Figure 2 – Transformers architecture.	24
Figure 3 – Mistral 7B benchmark.	26
Figure 4 – Architecture Mixtral 8x7B.	27
Figure 5 – <i>Benchmark</i> Mixtral 8x7B.	27
Figure 6 – Retrieval approach and generation with <i>seq2seq</i> and <i>Maximum Inner Product Search</i> (MIPS).	29
Figure 7 – Naive RAG schematics.	30
Figure 8 – Indexing step of RAG.	32
Figure 9 – Augmentation and Generation steps of RAG.	33
Figure 10 – Simplified diagram of data extraction.	34
Figure 11 – Simplified diagram of different modules of RAG solution.	39
Figure 12 – Representative diagram of failures and troubleshooting process. . .	42
Figure 13 – Example of a work order in the <i>Total Productive Maintenance Center</i> (TPMC) interface.	44
Figure 14 – Example of a long down file.	45
Figure 15 – Main page of the User Interface.	48
Figure 16 – User interaction with the RAG system for one work order.	49
Figure 17 – Summarization of example work order.	50
Figure 18 – List of part replacements for work order 1064326372.	51
Figure 19 – List of part replacements for work orders filtered by the term ‘faute 92’. .	52
Figure 20 – Root cause for work order 1064326372.	54

LIST OF TABLES

Table 1 – Metrics and weights used for qualitative evaluation.	37
Table 2 – Proposed Scoring Scales.	38
Table 3 – Generation Parameters for the Transformers Library.	40
Table 4 – Project Timeline.	43
Table 5 – Collected metrics during test session.	56

LIST OF ABBREVIATIONS AND ACRONYMS

AI	<i>Artificial Intelligence</i>
API	<i>Application Programming Interface</i>
AR	<i>Augmented Reality</i>
BART	<i>Bidirectional and Auto-Regressive Transformers</i>
BERT	<i>Bidirectional Encoder Representations from Transformers</i>
CBM	<i>Condition Based Maintenance</i>
CM	<i>Corrective Maintenance</i>
CNN	<i>Convolutional Neural Network</i>
GenAI	<i>Generative Artificial Intelligence</i>
GPT	<i>Generative pre-trained transformer</i>
I4.0	<i>Industry 4.0</i>
I5.0	<i>Industry 5.0</i>
IIoT	<i>Industrial Internet of Things</i>
IoT	<i>Internet of Things</i>
LLM	<i>Large Language Model</i>
MIPS	<i>Maximum Inner Product Search</i>
NLP	<i>Natural Language Processing</i>
NSM	<i>Non Scheduled Maintenance</i>
PdM	<i>Predictive Maintenance</i>
PM	<i>Preventive Maintenance</i>
RAG	<i>Retrieval-Augmented Generation</i>
RAGAS	<i>Retrieval Augmented Generation Assessment</i>
REALM	<i>Retrieval-Augmented Language Model Pre-Training</i>
RNN	<i>Recurrent Neural Network</i>
RUL	<i>Remaining Useful Life</i>
RxM	<i>Prescriptive Maintenance</i>
TPMC	<i>Total Productive Maintenance Center</i>
VR	<i>Virtual Reality</i>

CONTENTS

1	INTRODUCTION	14
1.1	OBJECTIVES	17
1.2	DOCUMENT STRUCTURE	17
2	CONTEXT	18
3	LITERATURE REVIEW	19
3.1	INDUSTRY 4.0 AND 5.0	19
3.1.1	Augmented Maintenance	19
3.2	GENERATIVE ARTIFICIAL INTELLIGENCE FOR AUGMENTED MAINTENANCE	22
3.3	LARGE LANGUAGE MODELS AND GENERATIVE ARTIFICIAL INTELLIGENCE	23
3.3.1	Mistral 7B Instruct	25
3.3.2	Mistral 8x7B Instruct	26
3.4	RETRIEVAL-AUGMENTED GENERATION (RAG)	28
4	MATERIALS AND METHODS	31
4.1	LOCAL NAIVE RAG	31
4.2	EXTRACTION, TREATMENT AND DATA STORAGE	32
4.3	<i>PROMPT</i> BUILDING	36
4.4	RETRIEVAL AND GENERATION EVALUATION	36
4.5	DEVELOPMENT ENVIRONMENT	38
4.6	CLASS STRUCTURE AND SOFTWARE MANAGEMENT	39
4.7	GENERATION PARAMETERS	40
5	SOLUTION PROPOSITION AND REQUIREMENTS	41
5.1	USER CONTEXT AND SOLUTION BOUNDARIES	41
5.2	PROJECT MANAGEMENT AND CHRONOLOGY	42
5.3	PROJECT'S DATA AND STRUCTURE	44
5.4	DATA ANALYSIS AND EXPLORATION	45
6	RESULTS AND DISCUSSION	47
6.1	TPMC MODULE	47
6.2	LONG DOWN MODULE	55
6.3	TOOL'S EVALUATION BASED ON USER'S FEEDBACK	55
7	CONCLUSION	57
	References	58

1 INTRODUCTION

In the industrial context, maintenance refers to the systematic actions taken to preserve and restore machinery, ensuring optimal performance and minimizing downtime during the production process. This availability can be achieved through interventions, whether in the case of occasional equipment failure or through techniques aimed at prolonging their useful life (SIMEU-ABAZI; SASSINE, 2001). However, these activities require the use of human and material resources, increasing its cost. Leading up to 40% of the total cost of the production (YUNIAWAN; ITO; MOHAMAD, 2013), (VISHNU; REGIKUMAR, 2016), (LEMES; HVAM, 2019). Therefore, it is important to seek ways to reduce maintenance costs to ensure the competitiveness of companies, foster innovation, preserve natural resources.

Maintenance encompasses more than just ensuring machine availability and boosting production; it also involves improving reliability, safety, and overall efficiency. Reliability refers to the machine's ability to perform its intended function consistently over time, while availability denotes the proportion of time the machine is operational and accessible for use ((SMITH, 2021);(MOBLEY, 2011)). These techniques also ensure the safety and well-being of operators and people present in the environment where the equipment is located, whether directly or indirectly. With the advancement of technologies and the concepts developed by social studies, the human being is increasingly placed at the center of production processes, as a fundamental agent and as the ultimate goal. Technology offers its support and enhances human capabilities, moving beyond the idea that the machine replaces man and making room for the new concepts of industry 4.0 and 5.0, where the human being is enhanced and assisted by technology (BAS VAN OUDENHOVEN PHILIPPE VAN DE CALSEYDE; DEMEROUTI, 2023).

The most basic and fundamental maintenance technique is *Corrective Maintenance* (CM), which consists of intervening in the machine when it presents a critical failure, that is a malfunction or breakdown of an equipment that renders it non-operational, halting production processes and requiring immediate repair or replacement to restore functionality (MOBLEY, 2011). Critical failures, which result in the total stoppage of the machine or the impossibility of performing the assigned task, and non-critical failures, which degrade performance, but do not prevent production, in contrast to this strategy, *Preventive Maintenance* (PM) focus on increasing the useful life of machines and their components, maintenance actions can be carried out periodically, such as the replacement of parts, cleaning, and inspections (POÓR; BASL; ZENISEK, 2019).

Maintenance 4.0 (also known as Smart Maintenance or Augmented Maintenance), refers to integration of *Internet of Things* (IoT) and the *Industrial Internet of Things* (IIoT), *Artificial Intelligence* (AI), Digitalization, the Distributed Systems, Aug-

mented Reality (AR) and *Virtual Reality* (VR), cloud computing, and Cybersecurity. The definition of the concept of Industry 4.0 varies according to different authors in the literature, but these characteristics are generally considered essential for the creation of a Maintenance 4.0 ((ACHOUCH et al., 2022); (NARDO et al., 2021); (IUNG; LEVRAT, 2014)).

The implementation of the pillars of Industry 4.0 mentioned above paves the way for other maintenance strategies. Thanks to the availability and the increasing quality of data, obtained especially through connected sensors and structured databases, as well as the current computing power that allows the implementation of data-based models, new maintenance strategies can be considered.

Another strategy is the *Predictive Maintenance* (PdM), consisting on the detection of critical failures and degradation of machine performance based on data, physical models, or a hybrid model is a reality and allows decision-makers to intervene in equipment before production or safety is compromised (KELEKO, A. et al., 2022). The most commonly used sensors in this strategy are vibration, acoustic, temperature, and pressure types, as previously mentioned (ZHANG; YANG; WANG, H., 2019a).

However, the implementation of this strategy requires a large amount of data for data-driven solutions, as well as complex models for solutions based on physical models, which is not always an option. It is also important to note that in contexts where machines are at the beginning of their life cycle or where the conditions of machine use do not allow for a significant amount of failure data, the construction of data-based solutions becomes relatively complex (YAN et al., 2017), (MOYNE; ISKANDAR, 2017a). The main impediments to implementing Predictive Maintenance (PdM) at STMicroelectronics are the internalization of maintenance, lack of failure data for modeling, and the unpredictability of infrequent failures.

Predictive maintenance (PdM) aims to identify failures before they occur. To prevent such failures from happening and to keep the machine's production and health status at an acceptable level, decisions must be made. From the moment the potential failure is signaled to the decision-maker (maintenance management and responsible for manufacturing site), a sequence of events can occur, and a technical intervention is then necessary to execute the most appropriate action plan for each case (ARENA, S. et al., 2022).

Artificial intelligence techniques assist in decision-making about what to do and how to do it for each case and situation, based on knowledge from technical documentation, intervention history, machine data, and others. This maintenance strategy is called RxM, which combines the knowledge of the human expert with the data processing power and the statistics of artificial intelligence (MATYAS et al., 2017).

As previously mentioned, the human being is at the center of the current industry, a concept so important that it has created a new denomination for the industry called

industry 5.0, which emphasizes the interface and collaboration between man and machine (BARATA; KAYSER, 2023), (REISCHAUER, 2018). This interface is complex and has been the subject of numerous studies. One of the great technological advances that has the potential to revolutionize this human-machine interface is the use of *Large Language Model* (LLM) and *Generative Artificial Intelligence* (GenAI) (JEONG, 2023).

The application of GenAI in the industry can take different directions, such as the analysis of sensor data, integration with documentation systems, or the development of cognitive assistants that allow operators and maintenance personnel to converse with the system in a way similar to a conversation between humans. One of the difficulties in implementing PdM is the users' acceptance of the intelligent systems' suggestions about what to do, often due to the lack of interpretability of the artificial intelligence models used. The use of GPT models, although dating back to 1992 with the appearance of the term, has seen greater implementation since the development of the *encoder-decoder* architecture presented by the *Brain* research group and the *Research* group at Google (VASWANI et al., 2017).

The impact of GenAI promises to revolutionize the way we interact and the entire system of work and production. Having the ability to converse with such an intelligent system is just one of the many capabilities of GenAI. Studies also show that the use of GPT as a question-and-answer assistant in specific domains is limited, given that the number of model parameters is limited, even if it reaches hundreds of billions, and that the data used for training such models are also limited (ZENG et al., 2023).

The data used for training are called the parametric knowledge of the GPT model, while the data that were not included in the training of such a model are called non-parametric. The lack of parametric knowledge to provide certain answers to questions posed by users can lead to what is called hallucination, where the model generates a response that is not related to the question asked. In the case of domain-specific questions, this problem is even more pronounced and complicates the implementation of cognitive assistants for such tasks, like smart maintenance. The limitations of LLM in terms of parametric knowledge can also be due to outdated information and reasoning limitations of the model, which contribute to the generation of hallucinations. To deal with this issue, techniques such as fine-tuning and retrieval-augmented generation (RAG) have been developed, with the latter being the most promising in terms of implementation cost and positive results.

The concept of the RAG technique is based on the work presented by (LEWIS, P. et al., 2021) in which the authors improved the response capabilities of a conversational GPT by providing documents containing non-parametric knowledge, in a manner similar to open-book tests where humans can consult documents. The technique was evaluated with question-and-answer datasets and generated positive results (higher accuracy, adherence to facts, and reduction of hallucinations) compared to models

without access to non-parametric knowledge (closed-book test).

With the RAG technique, cognitive assistants with access to technical documents and maintenance records can be of great utility for carrying out maintenance tasks on the ground, thereby enhancing the capabilities of the human being, whether they are a maintenance technician, a decision-maker, or a machine operator. This opens the way for the potential development of such an assistant and for the evaluation of its impact on user experience during its use.

1.1 OBJECTIVES

The main goal of the project is **to reduce the variability of downtime and enhance factory performance, the project aims to develop a conversational cognitive assistant that will automate the problem-solving process, streamlining diagnosis and action.** The assistant will use the RAG technique to access databases and provide accurate and relevant information to the user, assisting in problem identification and decision-making to solve it. Additionally, the project considers generating technically document and structure the implementation of the developed software in production.

1.2 DOCUMENT STRUCTURE

This document is structured as follows:

In Chapter 2, the context of the company STMicroelectronics, its production process, and the social and technological impact in the region where the internship was carried out, as well as in the world, are presented.

In Chapter 3, the literature review and the state of the art in the different key points of the research are presented.

Chapter 4 addresses the methodology implemented in the project and its requirements.

In Chapter 5, the project requirements are detailed and what was developed to meet them is described.

In Chapter 6, the results obtained during the project are presented, as well as the progress of the project compared to what was initially planned.

Finally, in Chapter 7, the conclusions of the project, an analysis of the impact of what was developed, and the possibilities for future work in the same area are presented.

2 CONTEXT

STMicroelectronics is a Franco-Italian multinational semiconductor manufacturer, formed in 1987 by the merger of Thomson Semiconducteurs and SGS Microelettronica. The company designs, manufactures, and sells integrated circuits, serving three main markets: smart mobility, power electronics, and the Internet of Things (IoT). With a global presence, STMicroelectronics employs over 50 thousand people and has R&D and manufacturing centers in Europe and Asia. In 2022, the company generated more than 16 billion dollars in revenue, positioning it among the top semiconductor companies in the world.

The company maintains close ties with the Electronics and Information Technology Laboratory (LETI) in Grenoble, France, which has resulted in numerous joint industrial and scientific projects. STMicroelectronics is dedicated to innovation, investing about 12% of its 2022 revenue in R&D and holding approximately 19,500 active or pending patents. The company serves a wide range of markets, including automotive, industrial, aerospace, and consumer electronics, with notable customers such as Apple, Bosch, Tesla and SpaceX.

The internship was carried out at STMicroelectronics, in the central MDS team in Crolles (*Manufacturing Data Science*) which develops data-based solutions using techniques such as artificial intelligence, data science, and machine learning. This was also in collaboration with the research and development team, thereby contributing to the company's ongoing commitment to innovation and technological advancements in the semiconductor industry.

During the project, the goal was set to identify a use case with real maintenance data and users who could inform their needs for a tool and, subsequently, evaluate the results obtained by the solution. For this purpose, we selected the maintenance team of the Rousset manufacturing plant, located in the south of France. In this context, when a machine requires intervention, there are two levels of maintenance professionals who can act to resolve the problem: the first-level technician and the specialist. The specialist is called when a problem with a machine persists even after interventions by first-level technicians. The objective and focus of the work are centered on the specialist's interface with the machine and the process carried out, until then, manually in a conversational manner, using the proposed tool.

3 LITERATURE REVIEW

This chapter aims to conduct a systematic literature review addressing the main topics of the project, thereby providing a solid theoretical foundation for the identification of the most pertinent methodology for the proposed case study and its implementation.

3.1 INDUSTRY 4.0 AND 5.0

The concepts of Industry 4.0 (I4.0) are varied and can change depending on the author and the era. Among the different conceptualizations at the base of I4.0, some characteristics are fundamental; cyber-physical systems, automation, and data flow. Within the technologies that emerge from the advancement of computing and microelectronics, additive manufacturing, augmented reality, process simulations, autonomous robots, IoT and IIoT, data analysis, cloud computing, cybersecurity, and digitalization stand out (BUTT, 2020).

The interface with the human operator on the factory floor of a smart industry is fundamental and has become the center of attention for current research and technological developments. The idea is no longer to replace human workers in mass, but rather to enhance their capabilities and enable machine-assisted work. The construction of software robots (*Softbots*) and intelligent assistants that are more human-like, facilitating the human-machine interface, has been explored as an innovative and fundamental concept for the new stages of the industry (RABELO; ROMERO; ZAMBIASI, 2018).

In convergence with the idea of the augmented human, the concept of Industry 5.0 (I5.0) emerges, expanding the boundaries of technological development focused solely on production to a productive system centered on people and society, taking into account the social and environmental impacts of the implementation of new technologies (XU et al., 2021), (EUROPÉENNE et al., 2023). In this context, smart maintenance gains prominence as one of the application areas of these technologies, aiming to optimize maintenance processes and improve the reliability and availability of equipment.

3.1.1 Augmented Maintenance

Smart maintenance (or Augmented Maintenance) in the context of I4.0 and I5.0 can take on different aspects and encompass various tasks and technologies to assist decision-makers, technicians, engineers, and supervisory systems. Automatic monitoring of resources offers agility and robustness in analyzing the lifespan of equipment, which is fundamental for PdM. In the phase of developing solutions for maintenance, it is also essential to center the human being in the concepts and to consider the acceptance by end-users and comfort during use. The development of cognitive assistants for maintenance is a promising idea explored in the literature, which can assist in

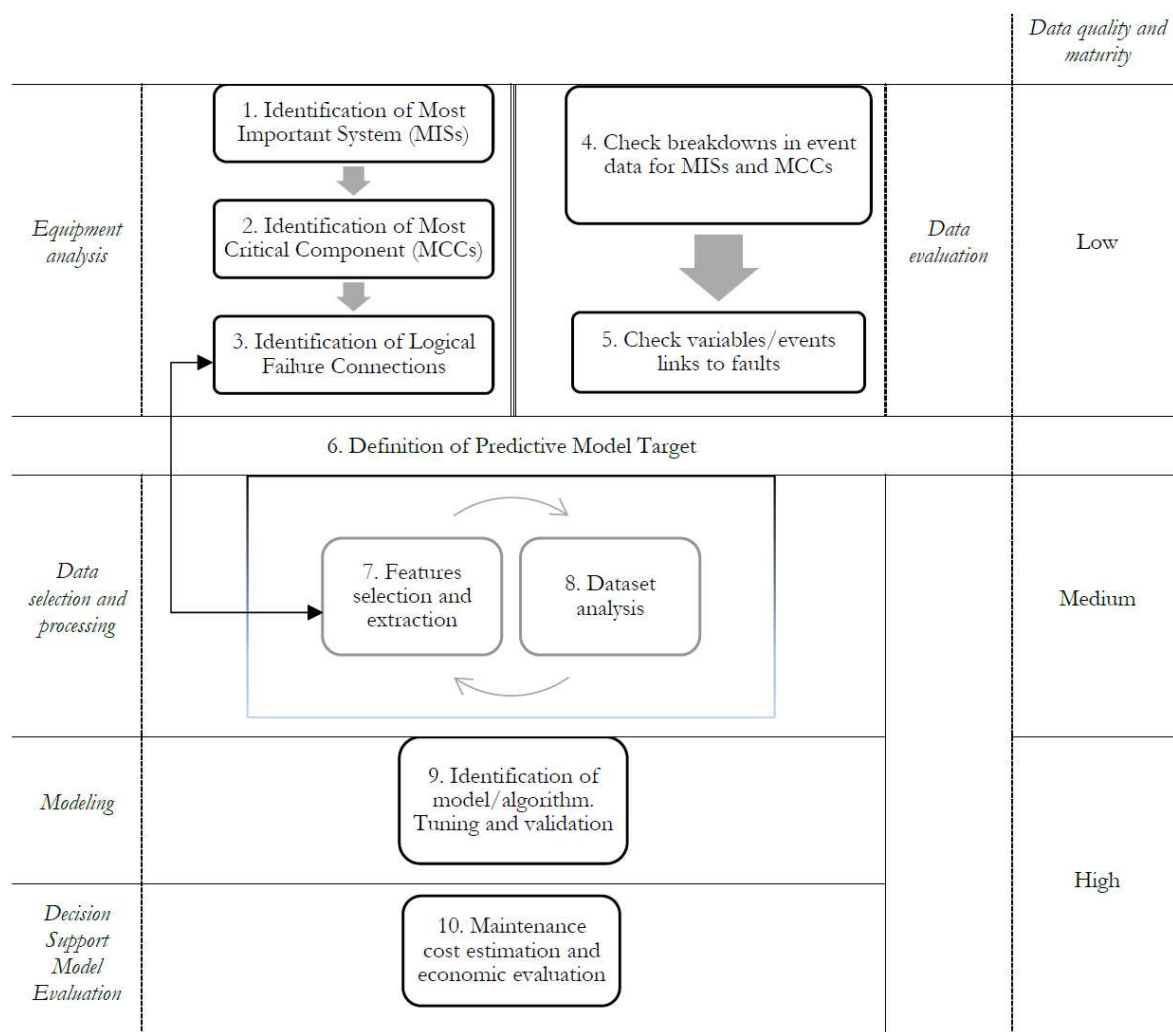
smart maintenance in its different dimensions, such as failure prediction, optimization of maintenance routines, and support for decision-making (VAN OUDENHOVEN et al., 2023).

In the field of smart maintenance, PdM receives greater attention. Technologies for remote asset monitoring, simulations, and digital twins allow health status prediction models of machines to contextualize decision-makers in real-time to decide the best moment for interventions on machines and systems (AFZAL et al., 2023), thus ensuring an increase in availability and reducing maintenance costs, which represent a significant portion of the total production cost (FLORIAN, Eleonora; SGARBOSSA, Fabio; ZENNARO, Ilenia, 2021). Intelligent sensing technologies also enable remote monitoring, an important concept for I4.0, providing a comparison between simulation data and the actual system, thus increasing the reliability of the prediction models used, as well as the quantity and quality of data to generate solutions called *data-driven* (ZHANG; YANG; WANG, H., 2019b), which can be integrated with the analytical power of GenAI (GOTO et al., 2023).

Three key aspects of smart maintenance are: Anomaly detection, failure prediction, and decision-making. These aspects are explored with different techniques and methodologies, depending on the context and available resources. Centering on the main goal, which is the increase in machine availability, reliability, reduction of losses and expenses, and increase in productivity, these goals are achieved either indirectly or directly. For such development and application, financial investments and skilled labor are required at different structural levels of the company, both in the training of personnel directly involved in the day-to-day maintenance and in the management and strategy dimensions of the company (BOUSDEKIS; APOSTOLOU; MENTZAS, 2019).

With the significant increase in computational capacity as well as the quantity and quality of data, machine learning techniques have gained greater prominence and have become the focus and trend of development in recent years. Such machine learning models have different applications in smart maintenance, such as *Condition Based Maintenance* (CBM), where the health of the equipment is predicted by data-driven machine learning models (CHIEN; CHEN, 2020) and indicators such as the *Remaining Useful Life* (RUL) of the equipment can be calculated, indicating the time to equipment failure and allowing the best moment for an intervention to be chosen, avoiding unnecessary downtime and also operations in states that decrease the quality of the processes and the overall life of the equipment. For such methodology to be applied, the data phase is essential, consisting of the acquisition, processing, and construction of the model that will assist in decision-making, becoming the stage that consumes the most time in machine learning projects and which is critical for the success or failure of the proposed solution (FLORIAN; SGARBOSSA; ZENNARO, 2020).

Figure 1 – Framework’s solution for the augmented maintenance.



Source: (FLORIAN; SGARBOSSA; ZENNARO, 2020)

The authors propose classifying smart maintenance solutions according to the maturity and availability of data for each context. Starting with relatively simpler tasks, such as fault and anomaly identification, and progressing to more complex tasks, such as decision-making support. What becomes clear from the presented framework is how important and complex the data flow within the enterprise system is and how massive the work to be done even before starting the technical solution is. Often, what happens is that the project exhausts its development budget in this initial phase or proves to be unfeasible due to the lack of data and essential structures needed to put a solution into production.

The application of artificial intelligence in the domain of smart maintenance depends, as seen, on the maturity of the data structure. When this maturity is present, more robust solutions with greater impact and value can emerge, thanks to the massive amount of available data and its respective diversity (MOYNE; ISKANDAR, 2017b). Real-time solutions for PdM with the integration of *Deep learning* for sensor data pro-

cessing and applications in classification, pattern detection, and prediction of numerical indicators are becoming increasingly a promising theme in the area of smart maintenance (KELEKO, A. T. et al., 2022). What is still being explored with the recent advances in the field of artificial intelligence is the part of decision-making support, which has proven to be a complex problem and can be approached in different ways and aspects.

Studies have investigated and proposed decision-making strategies based on the prediction of parameters in decision trees to decide the best maintenance strategy to be applied, with the focus on reducing the cost of operation and increasing the lifespan of the analyzed equipment (ARENA, Simone et al., 2022).

With the popularization of the Chat GPT solution from the company *OpenAI*, studies and applications of industrial solutions using generative artificial intelligence have increased considerably and have shown promising results for the area of smart maintenance, which will be addressed in the next session.

3.2 GENERATIVE ARTIFICIAL INTELLIGENCE FOR AUGMENTED MAINTENANCE

The application of GenAI in various industrial sectors has grown significantly, one of the main reasons for this is the availability of conversational interaction models (capable of interacting in a human-like manner) to the general public. Although it is a recent combination (GenAI and Industry), due to its relevance and popularity, the topic has been investigated in different ways, allowing the creation of new solutions for different manufacturing interfaces. This is even more evident when talking about I5.0, with the human being at the center of technological solutions, enhancing their capabilities beyond what can be achieved without such tools. An important factor for the success of the developed solutions is the acceptance of the end-users, who face limitations due to not having knowledge about how to correctly use the systems, understand their limitations, and also understand the process of information generation, a fundamental field of Comprehensive Artificial Intelligence. There is a demand for new technologies in different formats, and some can be cited, such as quality control, demand forecasting, assistance in operational procedures, product design, and customer relationship (DOANH et al., 2023).

From the applications enabled by generative artificial intelligence, the optimization of the human-machine interface is one of the most promising, providing more dynamic and fluid interactions between users and technology. For example, the *ChatGPT* tool is being widely adopted in customer interaction sectors and presents commercial solutions with great potential. The principle of using such an interface is to automate manual processes, facilitating the work of human beings without compromising the quality of the information flow. Two main advantages can be highlighted in the use of this tool: the massive amount of data used in its training (far superior to what a human being is capable of processing) and the flexibility to develop applications according to the need

and context. The tasks currently investigated that can be performed using this technology include virtual training of new employees, automatic listing of items and scheduling, big data analysis, creation of technical documentation, action planning, assistance in decision-making, problem-solving, among others (JAVAID; HALEEM; SINGH, 2023).

In addition, there are other techniques focused on maintenance that deserve attention. For example, implementation in the context of predictive maintenance has been gaining popularity in the industry, allowing for the prediction of failures and avoiding unnecessary downtime (CHOMKLIN; JAIYEN; WATTANAKITRUNGROJ, 2023). Moreover, anomaly detection and failure analysis using GenAI have also been widely studied ((GU et al., 2024), (EZUKWOKE et al., 2021)). Another important area of application is the assessment of equipment health through the analysis of maintenance logs, which allows for identifying trends and predicting possible future issues (BHARDWAJ; VEERAMANI; ZHOU, 2023).

GenAI has also been used for employee training, providing virtual simulations and interactive environments that improve learning and knowledge retention (FREIRE et al., 2024). Furthermore, the use of cognitive assistants for maintenance problem-solving stands out, which can provide real-time technical support and assist in decision-making ((MANTRAVADI; JANSSON; MØLLER, 2020), (PEREIRA et al., 2023), (BELLETIER et al., 2021), (TAIWO et al., 2024)).

3.3 LARGE LANGUAGE MODELS AND GENERATIVE ARTIFICIAL INTELLIGENCE

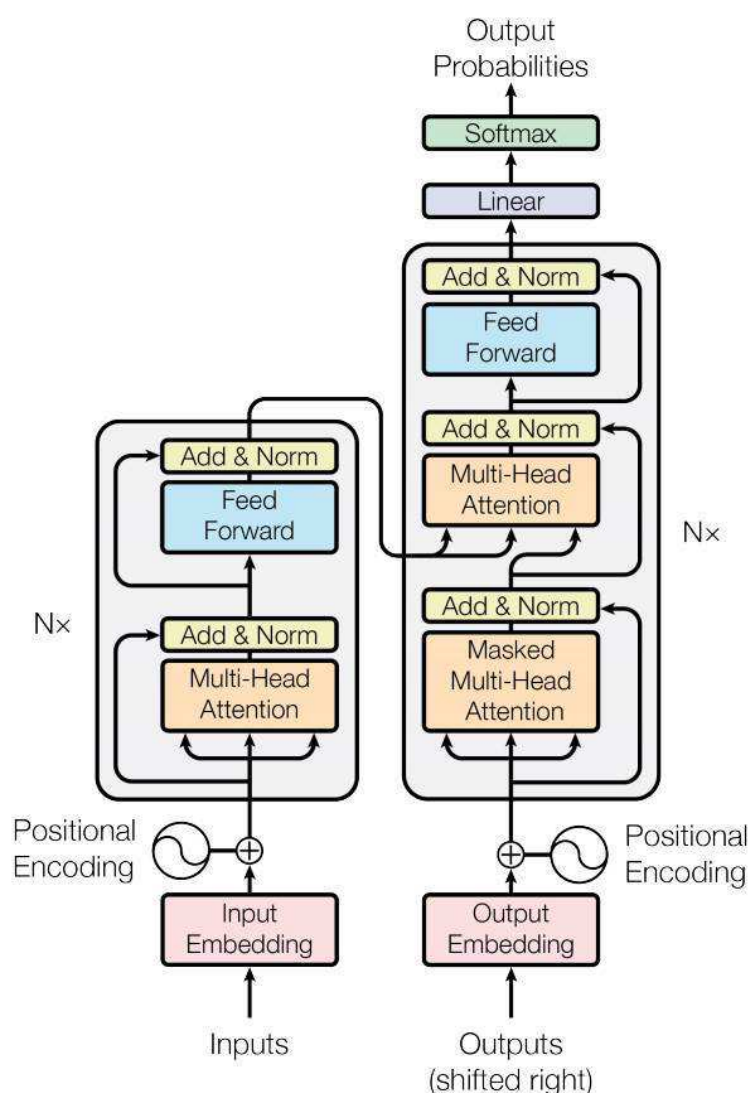
Large Language Models (LLMs) are capable of processing enormous amounts of data and providing responses in natural language. The use of LLMs in different fields of work has been investigated, and their potential lies precisely where human limitation is found: in the ability to process massive amounts of data and have simultaneous access to different sources to generate responses. The revolutionary mechanism behind these models, which enabled the expansion of research and development in the area, is that of the so-called transformers. The work by (VASWANI et al., 2017) introduced the Transformer, a deep learning architecture that leverages the attention mechanism. This architecture determines the probability of each token (word or symbol) in a sequence by considering the relationships and dependencies among all tokens, rather than just the preceding one.

In the field of natural language processing, tokens are the fundamental units of text that serve as input for computational models. They can be entire words, individual characters, or parts of words, and the choice depends on the applied segmentation method. The process known as tokenization involves dividing the text into these units, preparing them to be analyzed or processed by *Natural Language Processing* (NLP) models.

Embeddings, on the other hand, are vector representations of tokens that en-

capsulate semantic meanings and contextual relationships. In contrast to sparse representations, such as one-hot vectors, where each token is represented in isolation, embeddings align tokens with similar meanings or contexts close to each other in a continuous, multidimensional vector space. These vectors are generated through learning, in a way that mirrors the use and meaning of words in the real world, which significantly improves the ability of machine learning models to perform various NLP tasks (MIKOLOV et al., 2013).

Figure 2 – Transformers architecture.



Source: (PYTORCH FOUNDATION, n.d.)

The architecture of *Transformers* is structured around two main components: encoders to process the inputs and decoders to generate the outputs. The core of this architecture is the *multi-head attention* mechanism, which enables parallel processing of input sequences, improving performance and scalability compared to previous

methods such as *Recurrent Neural Network* (RNN) and *Convolutional Neural Network* (CNN). The encoder and decoder blocks are composed of attention sublayers and *feed-forward* networks, each followed by a layer of normalization and residual connections to facilitate the flow of gradients during training. In addition, position *embeddings* are needed for providing sequential context, allowing the architecture to handle complex and extensive contexts. *Transformers* are fundamental for advancement in NLP tasks, serving as the foundation for sophisticated language models such as *Bidirectional Encoder Representations from Transformers* (BERT) and GPT, thanks to their efficiency in parallel training and transfer learning capabilities.

The concept of the size of a Large Language Model is closely linked to the number of parameters that compose it. Simplified, these parameters represent the knowledge embedded in the model during training, being stored in the weights of the neural network used to predict text *tokens*. Generally, a model with more parameters has a superior capacity to store information compared to a smaller model. For example, *ChatGPT-4*, estimated at about 1.6 trillion parameters, surpasses in size models from other companies, such as those from Mistral AI, which have versions of 7 billion, 8×22 billion (176 billion), and 8×7 billion (56 billion) parameters, the latter being compositions of refined base models, known as mixtures of experts. However, it is important to emphasize that the number of parameters in a model does not translate directly into a proportional improvement in the quality of responses. Although it can significantly impact, other factors, such as the quality of the training data and the algorithms used, also play a definitive role in the model's ability to generate coherent and relevant responses ((BROWN et al., 2020), (MA; FANG; WANG, X., 2023)).

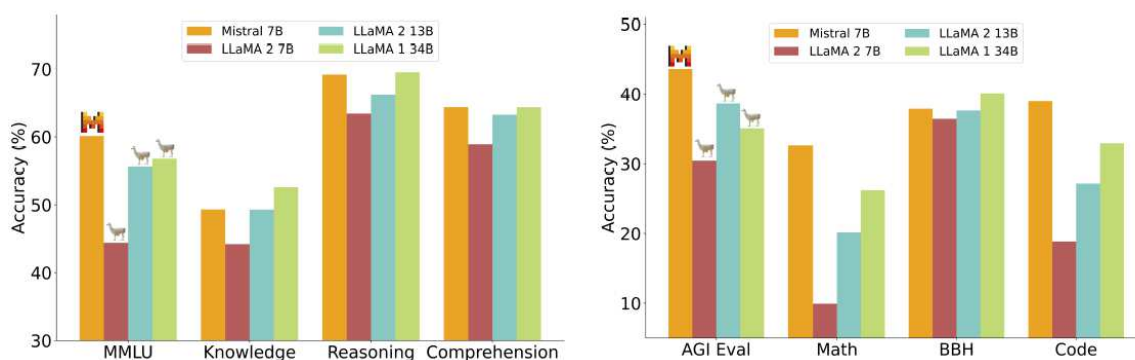
Different LLMs can be accessed through *Application Programming Interface* (API)s or hosted locally, especially in the case of open-source models. The advantage of researching open-source models lies in the transparency, collaboration, and diversity of applications they offer. On the other hand, models with greater generative capacity are often closed-source, which can hinder research and development due to data protection issues and business contracts. However, this can reduce infrastructure costs and development time, thanks to *pay-per-token* generated models and key sharing. During the current project, the focus will be on open-source models, such as those from the French company Mistral AI, which provide models with good generation results compared to others in the same category and allow for local development. Nevertheless, the use of inferences in closed-source models like *ChatGPT* through an API remains an open possibility for exploration.

3.3.1 Mistral 7B Instruct

The LLM *mistral 7B Instruct* from Mistral AI is an open-source model designed to be lightweight while maintaining or surpassing the performance of other open models

in question-answering systems and human knowledge tasks. The developers utilized two main attention mechanisms: *Sliding Attention Window*, which improved the model's speed by a factor of two, and the *Rolling Buffer Cache*, a token prediction method that stores a portion of the previously predicted tokens in a cache for efficient retrieval.

Figure 3 – Mistral 7B benchmark.



Source: (JIANG et al., 2023)

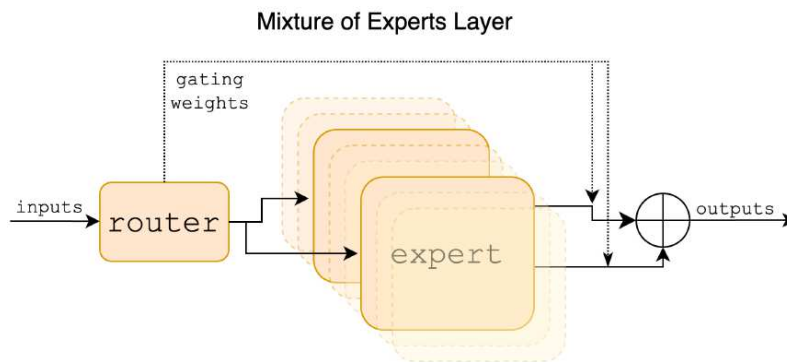
The model from Mistral AI proved to be superior to other models compared in the evaluated benchmark, especially in terms of reasoning and comprehension capabilities, which are essential for question-answer type tasks. The authors also noted that applying fine-tuning to the model to understand instructions (*instruct*) allowed for a qualitative improvement in the model's performance in terms of generating responses. (JIANG et al., 2023)

Another positive characteristic of the model is that it was also trained in the French language, which, due to the context of the present work, has a significant impact on the choice of which model to test and use in order to implement a solution based on natural language.

3.3.2 Mistral 8x7B Instruct

As previously mentioned in the section on the 7 billion parameter model, the 8×7 billion model combines expanded parametric knowledge with refinement using the same base model trained on datasets from various knowledge domains (JIANG et al., 2024) to perform specific tasks. This larger model leverages the increased number of parameters to enhance its ability to understand and generate complex patterns and nuanced information. By integrating diverse datasets, it achieves higher accuracy and reliability in specialized tasks, ranging from natural language processing to predictive analytics, thereby demonstrating significant advancements in computational efficiency and task-specific performance.

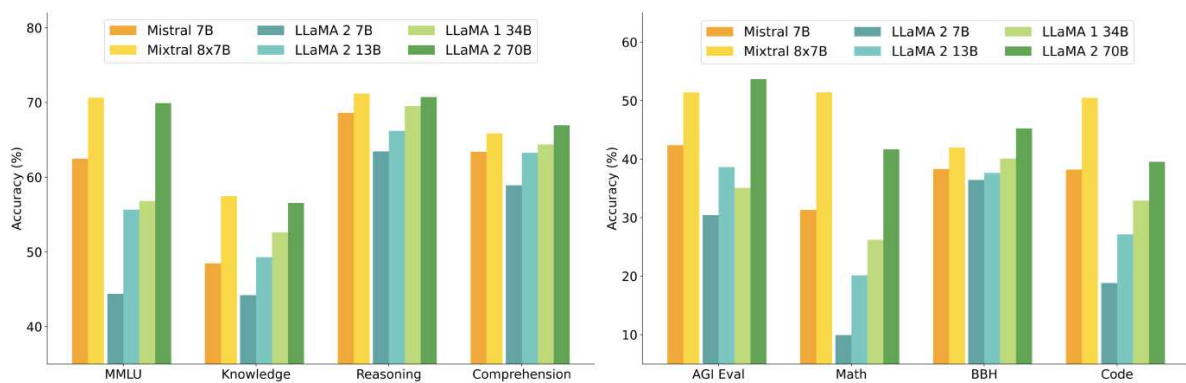
Figure 4 – Architecture Mixtral 8x7B.



Source: (JIANG et al., 2024)

In the architecture proposed by the authors, each user request is directed to two of the eight experts, determined by the *router* module. Then, the weighted responses from these experts are summed and presented at the model's output for decoding. This refinement approach resulted in a performance that was comparatively superior to that of other open-source and commercial models in evaluation metrics of public question-and-answer databases, along with refinement to receive instructions.

Figure 5 – Benchmark Mixtral 8x7B.



Source: (JIANG et al., 2024)

In the figure, the model's prominence in various selected metrics is evident, demonstrating its ability to handle intellectual tasks across different domains of knowledge. As mentioned earlier, due to the complexity and popularity of the topic, throughout the research and development of the project, other open-source models such as *Llama 3* (AI@META, 2024) were made available to the public, and their abilities proved to be equal to or greater than the Mixtral model in various categories. In some metrics, Llama 3, Mistral and Mixtral models even matched the performance of ChatGPT-4. However, it is necessary to focus on a specific set of features to clearly delineate the research, while remaining open to future developments.

Among the various tasks and model implementations, one methodology stands out for increasing the generative capacity of LLMs without applying fine-tuning: augmented generation by retrieval, which will be discussed in the next section.

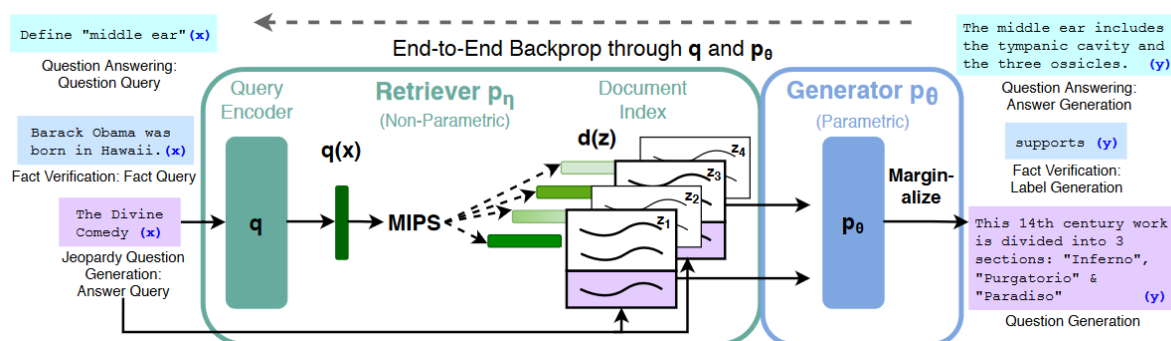
3.4 RETRIEVAL-AUGMENTED GENERATION (RAG)

The techniques aimed at improving the content generated by artificial intelligence are based on expanding the context within which the model must conceive answers. In this perspective, the most promising technique in terms of performance is the RAG technique, a continuation of previous works and similar to the *Retrieval-Augmented Language Model Pre-Training* (REALM) technique, which consists of a mechanism for retrieving documents and information from a given non-parametric database to enhance the responses of the LLM. More specifically, RAG retrieves documents and information from external sources, such as knowledge bases, databases, or the Internet, to improve the contextual understanding of the language model and generate more accurate and complete responses. REALM is a technique that improves the responses of language models based on documents selected by an information retrieval mechanism. The authors trained the information retrieval mechanism in an unsupervised manner to find the k most relevant documents based on the user's query. The authors utilized the MIPS technique in the information retrieval mechanism and evaluated the technique with open-ended question datasets. The results showed that the technique outperformed previous research and improved qualitative aspects such as interpretability and modularity. The authors also trained the information retrieval mechanism to penalize irrelevant information and reward useful information (GUU et al., 2020).

The RAG technique was proposed by a research group composed of *Facebook AI Research, University College London, and New York University* (LEWIS, P. et al., 2021). The primary goal of this technique is to enhance the contextual understanding of the LLM and enable it to generate precise answers, factual references, and reduce hallucinations caused by the lack of parametric knowledge and other limitations of the model. The RAG technique was proposed as an alternative to the fine-tuning method, which involves extending the training of a model on a specific dataset containing the necessary information to generate responses to user queries. This approach requires a significant amount of time and financial resources and still presents limitations in terms of parametric knowledge contained within the training dataset. The authors of the RAG technique claim that their models generate more specific, diverse, and factual language compared to a baseline parametric seq2seq reference for language generation tasks.

Figure 6 presents the concept of the technique and how user requests can be improved by incorporating non-parametric knowledge retrieved from documents as presented by the original authors of the technique.

The authors provide an overview of their approach, which consists of combining

Figure 6 – Retrieval approach and generation with *seq2seq* and MIPS.

Source: (LEWIS, P. et al., 2021)

a pre-trained retriever (query encoder + document index) with a pre-trained *seq2seq* model (Generator) and fine-tuning the end-to-end system. To identify the most relevant documents z_i for a query x , they use Maximum Inner Product Search (MIPS). For the final prediction y , they consider z as a latent variable and marginalize the *seq2seq* predictions given different documents (LEWIS, P. et al., 2021).

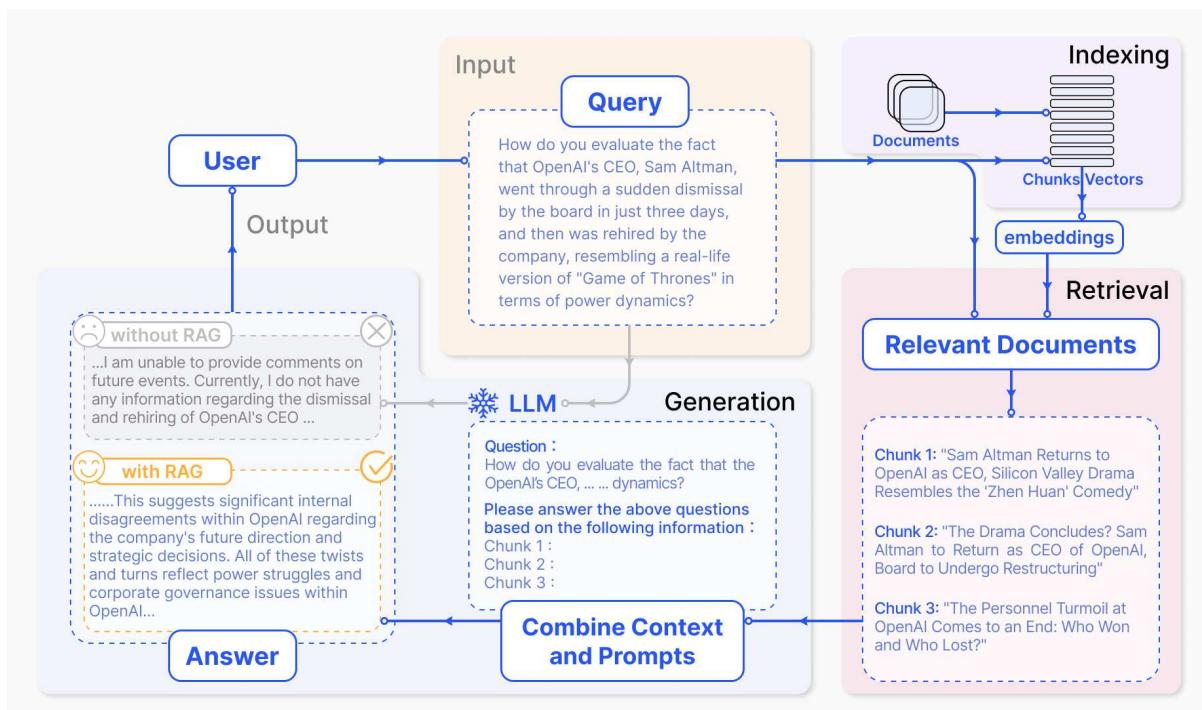
The RAG Framework includes retrieval, generation, and enhancement techniques that improve the model's ability to provide accurate and relevant answers. By accessing external data sources, RAG extends the context in which the model operates, resulting in more complete and contextually appropriate responses.

The authors provide an example where the dataset question is first passed through the request encoder. Then, the Maximum Inner Product Search (MIPS) mechanism is used to find the documents (indices) that most closely match the user's request, providing additional context to the question. The resulting information is then passed to the *Generator* model *Bidirectional and Auto-Regressive Transformers* (BART) (LEWIS, M. et al., 2019), which combines both parametric and non-parametric data to generate a more complete and coherent response based on the available information from the provided documents. To evaluate the performance of this tool, the authors used questions from public datasets and reference documents from the Wikipedia site. As mentioned earlier, the inclusion of non-parametric knowledge provided better responses with an improvement in hallucination and factual reliability compared to models that did not use this technique.

The study conducted by (GAO et al., 2024) mapped the advancement of the development of the RAG technique, which was divided into three paradigms: Naive RAG, Advanced RAG, and Modular RAG. The authors raise three important questions about the retrieval mechanism, including what content should be searched for, when it is necessary, and how to use it to extend the context for LLM. The use of structured text can significantly improve the context but may compromise efficiency and user acceptance in conversations.

Figure 7 shows the typical scheme of Naive RAG, which is the current standard of implementation, but not the most advanced.

Figure 7 – Naive RAG schematics.



Source: (GAO et al., 2024)

The Naive RAG is the classic methodology also known as *Retrieve-Read*. The downsides of this technique include the management of the data that make up the search base, the difficulty of retrieving relevant data with high precision, and the occurrence of hallucinations. The Advanced RAG was developed to overcome these issues, using techniques such as sliding window, fine segmentation, and metadata. Both the Advanced RAG and the Modular RAG allow for fine-tuning of the retriever, enabling the adaptation of the mechanism to domain-specific words of the work or company. The Modular RAG has become the standard implementation among researchers.

The authors address three problems that must be solved for the application of the RAG system: achieving correct semantic representations, aligning the semantic spaces of queries and documents, and aligning the output of the retriever with the preferences of the LLM. One way to optimize semantic representation is to consider the size of the segment used for information segmentation. Fine-tuning the *embedding* model (vector representation of the *tokens*) with domain-specific knowledge can also facilitate the process of information retrieval based on words and terms used in specific contexts (GAO et al., 2024).

4 MATERIALS AND METHODS

Given the nature of the data available for the project, as well as the choice to explore solutions in GenAI, the RAG *Naive* methodology was chosen as the most suitable. The idea of dividing the processing of information sources into different stages arose from interactions with users and detailing the fault diagnosis system, structuring the resources available for exploration and their implementation. The data available for the project consist of extractions from an *SQL* database containing entries related to service orders, as well as documents that record persistent failures over long periods of time, prepared by the maintenance team specialist for a group of equipment, aiming to map failures and propose solutions to the problems found.

Therefore establishing a rigorous methodology for evaluating a system's responses is essential to ensure accuracy, prevent critical errors, and foster user trust. Such a methodology not only aids in the initial user adoption by providing reliable information but also supports the solution's deployment and long-term maintainability. Whether through manual processes or automated strategies like *Retrieval Augmented Generation Assessment* (RAGAS) (framework based on critic LLM that will evaluate the answers from RAG systems), consistent evaluation is key to the system's continuous improvement and adaptability, ensuring it meets user needs effectively over time. The development location and the structures available for the implementation of the system were defined, enabling the realization of the project. The proposed system was modularized so that its code can be reused by other projects and teams, following the best practices for software development and technical documentation.

4.1 LOCAL NAIVE RAG

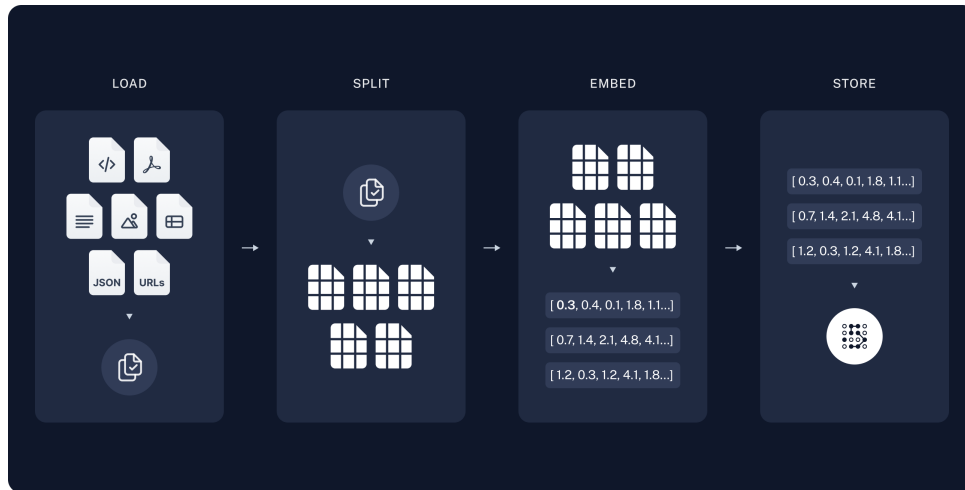
The retrieval-augmented generation system was implemented locally, prioritizing the security of the company's sensitive data, as well as allowing greater flexibility in proposing necessary modules as per user needs, low implementation and maintenance costs.

The integration of various maintenance data sources plays a decisive role in the contextualization and enrichment of the responses provided by the LLM. By combining information from service orders, failure logs, and other relevant sources, the system can better understand the specific context of operation and maintenance of the equipment. This not only increases the accuracy of the responses but also makes them more relevant to the end-users. In the specific case of STMicroelectronics, this customization of the cognitive assistant allows access to restricted information, essential for the maintenance team but not available to the general public (non-parametric knowledge).

As previously presented, the system has three main parts: information indexing, context augmentation, and generation. Each main part requires specific characteristics

for its implementation and proper functioning. Figure 8 shows the simplification of the information indexing stage.

Figure 8 – Indexing step of RAG.



Source: (LANGCHAIN, INC., n.d.)

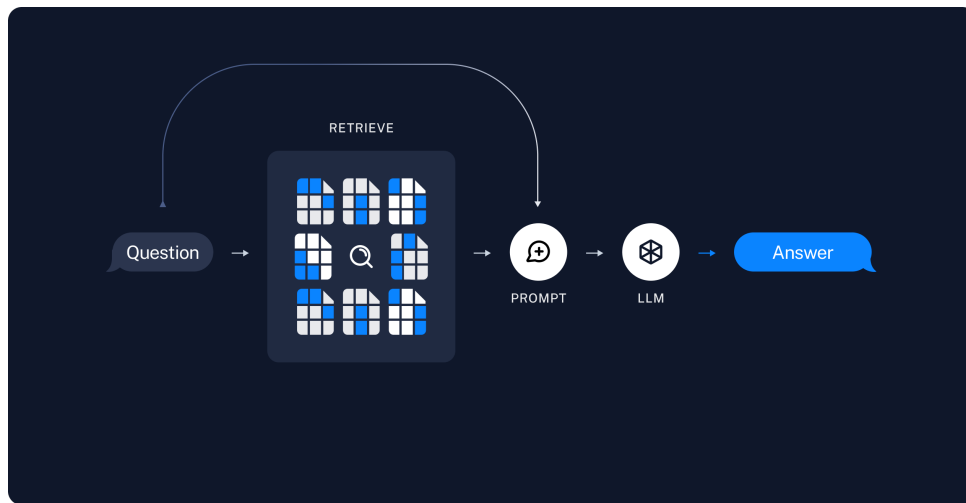
The first stage, called *load*, loads the documents in different formats (for example, *PDF*, *XLSX*, *DOCX*, *CSV*, etc.) into the system, performing the necessary data treatment for each document, considering its particularities. The next stage, called separation (or "*split*"), divides the textual content of the loaded documents in a way that preserves the context and the semantic relationship between the different pieces of information, limiting the number of *tokens* and the overlap of the chosen parts, according to the adopted strategy. The *embed* stage consists of transforming the "*chunks*" into vector representations, ending with the storage of these vectors. Once this stage is completed, the process moves to the stage of augmenting the request and then generating the response, as presented in Figure 9.

Given a user request, transformed into a vector representation (*embedding*), the retrieval module (*retrieve*) will search for the vectorized *chunks* of documents that are closest in Euclidean distance by similarity. With the recovered context and transformed back into text, the instructions to the model, called *prompt*, will be composed of a base instruction (*instruct*), the augmented context, and the user's request, then moving to the generation model, in the current case a LLM, and finally, returning a response to the user.

4.2 EXTRACTION, TREATMENT AND DATA STORAGE

After initial contact with the maintenance team of one of STMicroelectronics' factories (Rousset), data from machines located in an area called *Chemical Vapor Deposition* were selected by the users themselves according to the priority they determined to be sufficient and necessary to diagnose and make decisions to put the

Figure 9 – Augmentation and Generation steps of RAG.



Source: (LANGCHAIN, INC., n.d.)

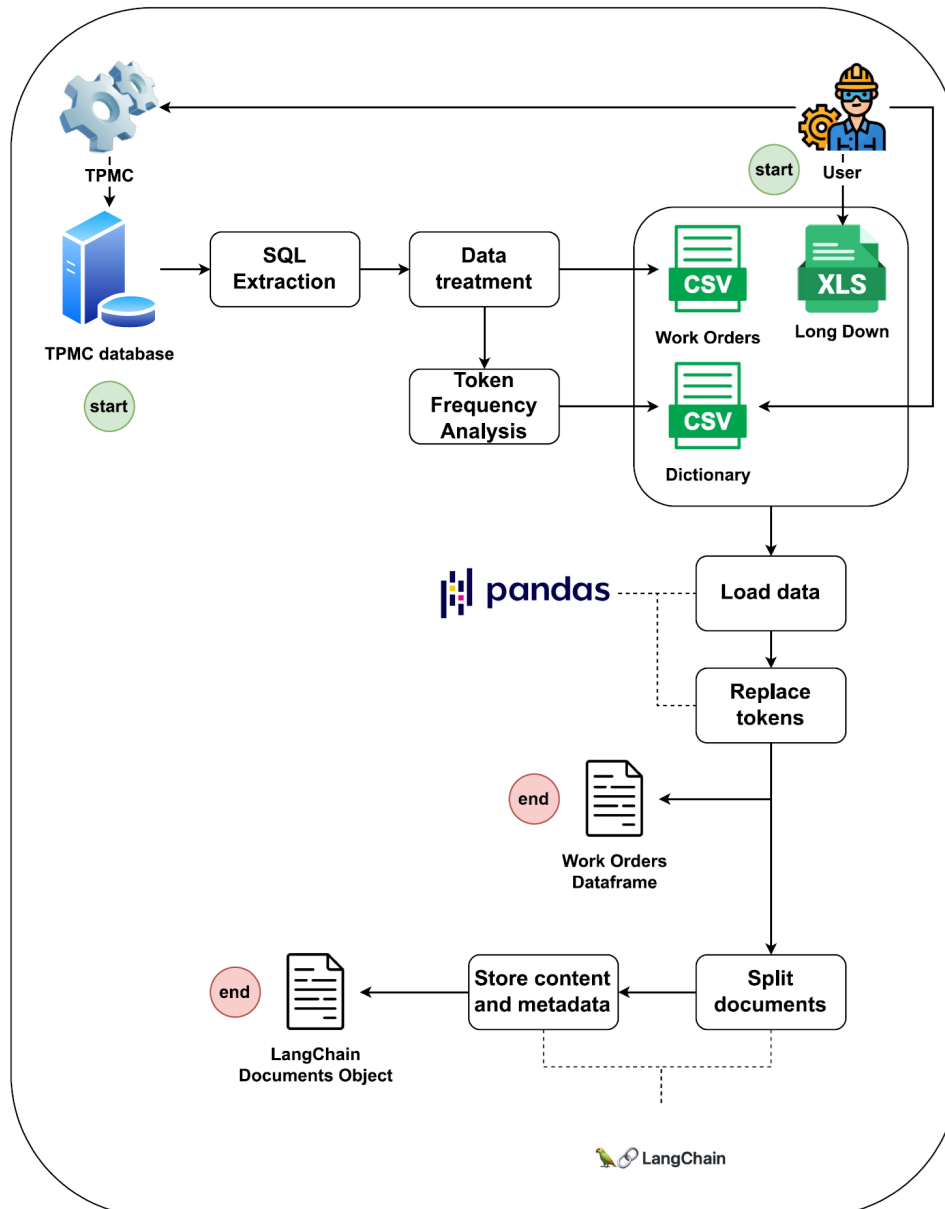
machine back into production. The data related to service orders, lessons learned between machine downtimes, and user and maintenance manuals have extensions such as *DOCX*, *DOC*, *PDF*, *CSV*, *XLS*, *XLSX*, *PPTX*, and *JPEG*.

The *long down* and TPMC documents have fields with entries by users, with some parts in French, some in English, and even in Italian, but mostly in French. With this information, a token substitution technique was proposed as part of the data processing. With the help of the data scientist responsible for analyzing the database and the maintenance manager, the thousand most frequent tokens in the comments of service orders in English and French were selected, and a short substitution was suggested to be performed at the time of data processing. A characteristic example of necessary substitution is the abbreviation of the French word '*chgt*', or '*changement*', which means replacement.

The importance of this step has become fundamental, as it was found that, given the use of abbreviations in French or jargon used by maintenance that are not generally known outside the context of STMicroelectronics, there is a high probability of hallucination at the time of generation. This occurs because the model will receive this token and, even if it does not have information from the context, will try to explain according to its parametric data and probabilities of meaning. Finally, 750 tokens were filtered and their substitutes were defined.

The file named *Dictionary* with the token replacement is loaded into the system (CSV format), along with the *long down* file (XLS and XLSX) and the service order file (CSV format). These data are processed by a Python class created to perform this task, which converts them into LangChain objects and applies the token separation strategy. A simplified diagram of the data extraction and processing is presented in Figure 10.

Figure 10 – Simplified diagram of data extraction.



Source: Author

The service order data are centralized in an *Oracle* database, where the extraction of cross-referenced information on corrective, preventive, and non-standard actions was carried out. With the data extracted from the database, two main tasks were performed: the compilation of different stages of the service orders (*ACTION*, *ANALYSIS*, *PASSDOWN* and *PLANNER COMMENTS*) and their chronological organization, which is important information for identifying which action led to the solution of a particular problem.

In parallel, using the Python library *NLTK* (*Natural Language Toolkit*), the manual inputs (maintenance comments) from the technicians were segmented into tokens and analyzed for frequency, selecting then the (1000) most frequent tokens. With this

information, it was possible to construct a replacement dictionary, in which the identified token (in English or French) as an abbreviation or maintenance jargon receives a standard English substitution, allowing the application of this method in the subsequent data processing stage.

Additionally, the extended failure file (*Long Down*), a document in table format filled out completely manually and without a predefined organization, was made available. With these three products from the initial extraction (compilation of maintenance orders, dictionary, and extended failure), the following steps can be performed by loading such data into the developed system, applying the token replacement from the dictionary data, and creating the database with processed text from the service orders and extended failures.

Utilizing the *recursive character splitter* method provided by the *LangChain* package, the text data is segmented based on a specific character count threshold to maintain the integrity and coherence of the information. The splitting parameter was set to 500 characters per segment, coupled with an overlap of 50 characters for each consecutive segment. This overlap ensures continuity and context retention across the segments, or *chunks*. Consequently, these *chunks* are transformed into *LangChain* Objects, encapsulating both the textual content and pertinent metadata, such as the source, page number, and line information. The choice of the splitting parameters, particularly the 500-character limit, is often a balance between the granularity of the text and the need for context. Smaller *chunks* may lead to more precise analysis but can disrupt the narrative flow, while larger *chunks* might retain better context but could be unwieldy for processing. The 50 character overlap is a safeguard against the potential loss of meaning that might occur at the boundary of two *chunks*, ensuring that sentences are not cut off in a way that would impede understanding.

With the content of the created objects, the next step for saving such data for retrieval and prompt expansion for generation involves indexing, where the textual elements are vectorized using an *embedding* model. As previously presented, the *embedding* model will relationally transform the tokens present in the texts and construct a numerical relational representation among them.

The model chosen for this task was the *multilingual-e5-large-instruct*, refined to receive instructions (WANG, L. et al., 2024). It was selected because it is a public model, it shows good results in its metrics compared to other public models, and it is capable of working with several languages, as is the case with the present problem where French, English, and Italian are being used.

With the vectors of the extracted texts, it is then possible to create a vector database where the indices of the vectors are positioned in a space of dimension ($d=768$) and approximated by similarity. This allows the application of search methods where the Euclidean distances between the vectors are calculated in reference to a

particular input, in the present case, a user's question that is also vectorized.

4.3 PROMPT BUILDING

The construction of the *prompt*, follows a standard composed of: base instruction (or system instruction), added context, and the user's question. With these three elements, it is possible to obtain more complete responses with generative artificial intelligence models. The base instruction is essential, especially when using models refined to receive such instructions, like the *Mistral 7B Instruct* model. In this base instruction, one can textually apply the rules for generation that the model should follow.

The context is retrieved from the vector database, where the user's input, after being vectorized, is positioned within this vector space. The search then finds the (k) closest elements, based on the Euclidean distance between the other present indices. Thus, the user's input takes the value zero, and a *score* is assigned to the other returned indices; the closer to zero, the more related they are to the user's input.

Different techniques are applied to determine the best thresholds for retrieval. From testing, the parameters ($k = 20$) and threshold ($= 0.2$) were determined as the most suitable. Initially, all (20) indices (document *chunks* ingested in the previous step) are retrieved and subsequently filtered. If there are no documents remaining after the filter, the system then switches to simple response mode and generates content based solely on its parametric data and explaining to the user the lack of context and that it can't give the asked information.

The next step is to pass the complete prompt to the *tokenizer*. In the context of large language models (LLMs), a *tokenizer* is a component that transforms text into smaller units called *tokens*. After this division, the *tokenizer* maps each *token* to a unique numerical identifier, known as an *ID*, according to the model's vocabulary. This conversion is essential for the model to process and understand the input text efficiently and accurately. The *IDs* are then passed to the generation model to obtain the response in the desired format.

4.4 RETRIEVAL AND GENERATION EVALUATION

For the evaluation of RAG performance, the methodology that is most studied currently encompassing the ability of the retrieval system to identify relevant and focused context passages, the ability of the LLM to effectively utilize such passages in a faithful manner, and the quality of the generation itself, is referred to as (ES et al., 2023).

The authors propose that two agents be constructed, using different *GenAI* models: one that will answer the questions (response agent) and another that will create the questions and evaluate the responses of the first agent (critic). Generally, the

critic has access to the database that the response agent will use to retrieve context, and then creates a dataset of questions that should be answered by the first agent. In the research work that presented this methodology, there is a comparison between the models ChatGPT-3.5 (response agent) and ChatGPT-4 (critic).

In the current work, a similar method of evaluation is proposed. Given the limitation of resources (different types of models that can be used), the generation of questions is done by the future user of the tool (maintenance professional/expert), who also assumes the role of critic, evaluating the generated result in a qualitative manner.

Subsequently, a qualitative evaluation methodology was also developed in which the response is classified into five different levels, which are presented in Table 1.

Table 1 – Metrics and weights used for qualitative evaluation.

Description	Level
Completely false information (hallucination)	1
Partially false information	2
Partially complete information	3
Complete information	4
Complete and useful information	5

Source: Author

1. Response is completely false (hallucination).
2. Response contains parts with hallucinations but also correct information.
3. No hallucinations, but not all necessary information present (e.g., incomplete service order summarization).
4. All necessary facts presented correctly.
5. All correct information provided, includes useful reasoning (e.g., real cause of machine failure).
6. Evaluation process:
 - System exposed to specialist maintenance technician.
 - Technician evaluates each question-response pair based on context.

In a simplified manner, to calculate the average of the evaluations, the following formula is used $\bar{x}_p = \frac{\sum_{i=1}^n x_i}{n}$.

Normalizing the results between (0) and (100) allows for a more intuitive and comparable interpretation of the quality of the responses. This facilitates analysis and decision-making, as the values are on a familiar and standardized scale. Moreover, normalization helps to prevent distortions caused by differences in the evaluation

scales between the different levels of response. Thus, it becomes easier to identify patterns, trends, and areas for improvement, contributing to a more transparent and effective evaluation process. The weighted and normalized average is denoted by ($\bar{X} = \frac{\bar{X}_p - W_{min}}{W_{max} - W_{min}} \times 100$). In Table 2, the ranges and their quality interpretation for the average of the different experiments conducted are presented.

Table 2 – Proposed Scoring Scales.

Scoring Range	Description
0 → 20	Many hallucinations and false information
21 → 40	Some hallucinations, but still contains correct information
41 → 60	No hallucinations, but missing essential information
61 → 80	All necessary facts correctly presented
81 → 100	All correct facts and additional useful reasoning

Source: Author

The metric presented allows for the evaluation of the proposed solution and the investigation of possible future improvements such as additional context or a change in generation strategy.

4.5 DEVELOPMENT ENVIRONMENT

The development environment was proposed with a focus on the security of corporate data and the confidentiality of information related to production processes. For the use of large-scale language models, the use of graphics processing units (GPUs) is necessary, given the enormous amount of calculations performed for generation. With this in mind, access to the company's *computational farm* was provided, equipped with two graphics cards with (80) [Gb] of memory type VRAM, allowing to load models of limited size and perform inferences on local models. In parallel, in the second part of the project, GPU resources were made available on the Azure cloud. In both cases, the graphics cards used are of the *NVIDIA A100* type.

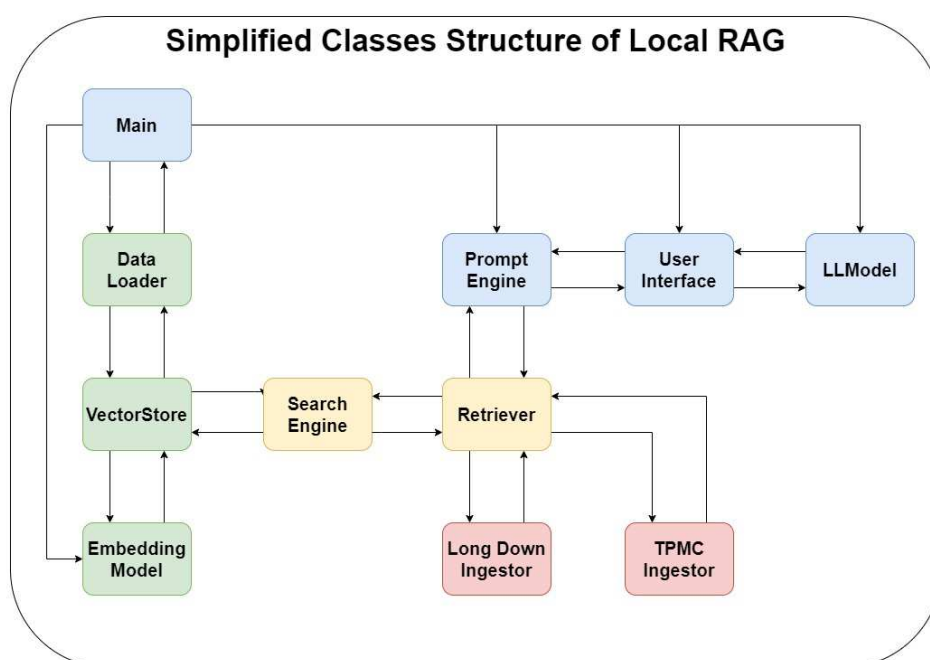
The code developed in Python language followed the internal docstring guidelines, the best practices of PEP8, was object-oriented, and had version control carried out by GIT.

As the goal is to develop a system applicable to the company's reality even after the project's conclusion, all the code was developed in a modular fashion. This allows other projects to use it and for development to continue in the best possible way for future improvements

4.6 CLASS STRUCTURE AND SOFTWARE MANAGEMENT

As mentioned earlier, the solution was designed in modules so that different parts of the system could be leveraged by other projects. Moreover, it is important to highlight the relevance of object-oriented programming in the construction of these modules. By adopting this paradigm, it becomes easier to maintain and develop additional modules, as classes provide an efficient way to encapsulate related data and behaviors. This promotes code reuse, increases readability, and facilitates maintenance by isolating functionalities into independent and interconnected units. The simplified structure of the developed classes is presented in Figure 11.

Figure 11 – Simplified diagram of different modules of RAG solution.



Source: Author

The central point of the solution is the main class, which creates objects of type *data loader*, *prompt engine*, *user interface*, and *LLModel*. Each of these objects plays a unique and important role in the solution. From the main module, the main parameters are loaded, such as the path of the generation and embedding models, the path of the data for loading, and other configurations necessary for the solution.

The data loader class iterates within the folders and subfolders of the provided path, recursively ingesting documents to create objects of type LangChain Document, and then creating the indices of the vector database with the generated embeddings.

The *Prompt Engine* class centralizes information that is retrieved by the *retriever* class, which searches for information from the *long down* module, *TPMC ingestor*, or through searches performed in the *Vectorstore* database.

The graphical interface, developed with the *User Interface* class, connects the other classes to the *LLModel* generation class, which loads the generation model, receives the *token IDs*, and returns the responses.

As the project progressed, the decision to separate the use of the two proposed modules was made in a way that facilitates the implementation of the tool and also adapts the project to the constraints of time and material resources.

4.7 GENERATION PARAMETERS

The output parameters of the generation model are presented in Table 3

Table 3 – Generation Parameters for the Transformers Library.

Parameter	Value	Description
top_k	50	Controls the sampling for each token, limiting the selection to the top_k most likely tokens.
top_p	0.9	Nucleus sampling, limiting the cumulative probability distribution to the top_p mass of probability.
temperature	0.6	Controls the randomness of the sampling process. Higher values lead to more random outputs.
do_sample	<i>True</i>	If <i>True</i> , allows sampling from the distribution of the next tokens.
length_penalty	0.3	Factor applied to the length of the sequences generated during decoding.
max_tokens	32e3	Maximum number of tokens that the model can generate.
num_beams	10	Number of beams for beam search decoding.

Source: Author

The generation process in the Transformers library employs several key parameters to control the model's output. Firstly, 'top_k' limits token sampling to the 'k' most likely tokens, while 'top_p' restricts the cumulative probability distribution to a subset of the most probable tokens. 'temperature' adjusts the randomness of sampling, influencing the balance between diversity and determinism in the outputs. By activating 'do_sample', it facilitates sampling from the distribution of the next token, promoting diverse outputs. 'length_penalty' discourages longer sequences, favoring shorter and more concise outputs. 'max_tokens' imposes an upper limit on the sequence length, while 'num_beams' dictates the number of beams used in beam search decoding, a strategy that explores multiple paths to select the most probable sequences based on a scoring function. These parameters, together, govern the generation strategy, balancing randomness, diversity, and coherence in the model's outputs.

5 SOLUTION PROPOSITION AND REQUIREMENTS

This section aims to present the project requirements, outlined based on the identified needs and the selected methodology. The steps and sub-steps necessary to achieve the proposed objectives will be detailed, as well as the comparison between the planned and actual schedule. Furthermore, the available database, infrastructure, and analysis carried out will be discussed, including initial investigations into the possibilities of development within the current scenario, considering the available resources and deadlines. Finally, the importance of adaptability and acceptability of the tool by the end-user will be addressed, as well as their experience over time.

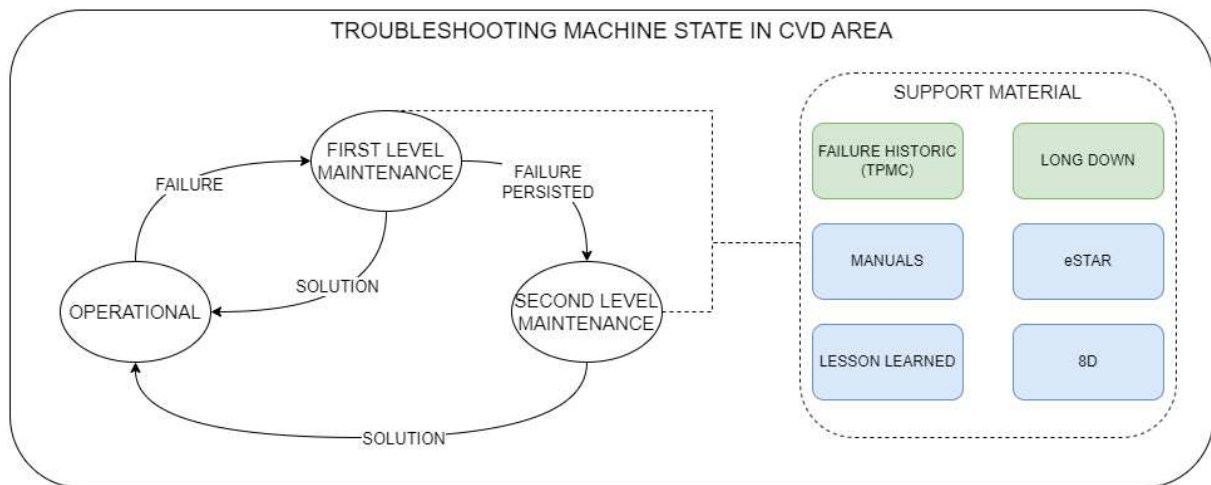
5.1 USER CONTEXT AND SOLUTION BOUNDARIES

After the presentation of the proposal and the scenarios available at the various production plants of the company *STMicroelectronics*, a team was investigated for data availability, motivation for implementing the solution, and capacity to assist in the exploration and implementation stages. The team responsible for maintaining the *Chemical Vapor Deposition (CVD)* machines at the Rousset plant was selected based on the previously mentioned criteria, as well as the maturity in understanding the need for the proposed tool and the experience of the factory floor technicians and new engineers in the context of the company. A critical information to choose the aforementioned plant was the presence of the Data Scientist from the Manufacturing Data Science team locally with access to the reacquired data and knowledge of the oracle base.

As discussed in the literature review section, maintenance costs represent a substantial part of the total production costs and, therefore, must be optimized to ensure the conservation of resources and the competitiveness of the company. One of the strategies to achieve these objectives is to reduce the time required to solve machine problems. Certain problems in the studied area are known as persistent failures, which persist even after several service orders are carried out to correct them and put the machine back into production.

In the current scenario of the studied team, when a failure or event requiring maintenance with machine downtime occurs, the first-level technician is responsible for the first step in problem resolution. He consults various sources of information, such as machine alerts and stoppage histories, and, based on these data and his experience, attempts to identify the root cause of the failure. If he is unable to resolve the problem, he escalates it to the second-level technician. The second-level technician, also known as a specialist, follows a systematic process of root cause investigation, going through different stages until finding a definitive solution that allows the machine to be put back into operation.

Figure 12 – Representative diagram of failures and troubleshooting process.



Source: Author

The diagram was developed after several interactions with the maintenance team, who shared their context and identified the most relevant documents for fault detection and the actions previously taken to solve similar problems. In addition to the primary sources of information, such as machine manuals and alarms, other secondary sources were considered, as detailed in the methodology section. It was decided to follow the same flow of the user, focusing on the interaction of the second-level technician during the exploration phase, while the tool was also developed to be used by other maintenance professionals.

Based on the RAG methodology, the goal is to develop a cognitive assistant to expedite the root cause investigation process of machine failures, eliminating long periods of machine unavailability, thus leading to better system linearity and providing useful information based on the provided context. Among the modules mentioned earlier, the failure history analysis module (TPMC) applies filters to the database extracted from maintenance comments (in natural language) and uses the selected information as context to summarize and respond to questions posed by the user, aiming to obtain correct, useful, and objective answers.

The second module developed performs context retrieval in the vector database, seeking similarity between the user's question vector and the representative vectors of the stored data. This approach extends the prompt provided to the language generation model, resulting in more informative and relevant answers. Subsequently, these answers are evaluated according to the metrics presented earlier in the tasks of summary and identification of solutions for persistent problems.

5.2 PROJECT MANAGEMENT AND CHRONOLOGY

The proposed project was divided into five macro stages.

1. Define theme, problem boundaries, stakeholders, challenges, methodology, and data specifications.
2. Plan data analysis, utilize end-user feedback, and develop software for Retrieval-Augmented Generation.
3. Apply methodology, document solutions, and prepare for production deployment.
4. Develop based on user feedback, refine for production readiness.
5. Organize documentation, present results, and conclude the project.

Particularly highlighted are the activities related to defining the use case, the limits of the proposed solution, the treatment of the available data, the choice of methodology based on literature and commercial solutions, as well as the validation with supervisors and end-users. Table 4 presents the expected actions and developments for project management as the releases occur throughout the period.

Table 4 – Project Timeline.

Objective	Start date	End date
Project setup and definition	5/2/2024	31/03/2024
Data ingestion development	31/03/2024	17/05/2024
Content summarizing	31/03/2024	31/05/2024
Alpha release	-	31/05/2024
User evaluation and improvements	31/05/2024	21/06/2024
Beta release	-	30/06/2024
Documentation	5/2/2024	26/07/2024
Results presentation	1/7/2024	26/07/2024
Final release	-	26/07/2024

Source: Author

The project startup phase is centered on the collection of diverse requirements and the exploration of possibilities for developing an augmented maintenance solution, utilizing resources such as data, the environment, and feedback from early-stage users. The objectives for this phase include defining the basic framework and project specifications as well as the data structures that will be employed.

In the subsequent phase of data ingestion development and content summarization, the goals are to construct a data ingestion engine, establish a filtering system tailored to the context, and develop modules for summarizing faults and long down documents. Additionally, efforts will be made to develop a retrieval engine for augmentation and a user-friendly interface.

These steps are critical for the development of the solution, as they rely on essential tools that the system must possess to implement the RAG methodology,

which includes the data ingestion pipeline and mechanisms for context-sensitive filtering based on user needs.

The final part of the project involves the alpha release of the solution to users, aiming to collect feedback for system improvement in the next phase. The beta release will focus on refining the solution and, based on evaluation metrics, modifying the approach to deliver a practical solution. Documenting and presenting the results are crucial, as they provide a synthesis of the development process, the value delivered, and the impact on the end-user.

5.3 PROJECT’S DATA AND STRUCTURE

After identifying the user requirements and establishing the boundaries of the solution, it was defined two primary sources of data: the TPMC database and the long down files. For the former, as detailed in the Materials and Methods section, the most pertinent data is derived from the natural language entries made by technicians during machine maintenance operations.

Figure 13 – Example of a work order in the TPMC interface.

							Type	When	Who	Step	Tasks	Comments
DGF03A	15 Mar 2024 14:59	laurent burtin	CAUSE RF-BIAS MATCH CAUSE RF-AUTRE	faute rf	DERIVE RANGE RF SIDE	NA	EVENT	15 Mar 2024 14:59	laurent burtin	Maintenance		COUNTER VALUES: Wafer_16k=1454.0; ACTUATOR=195116.0; ESC=40141.0; PA_TUBE_SIDE=38900.0; PA_TUBE_BIAS=21103.0; PMRF_TimeOut=90.0; Dome=189815.0; PA_TUBE_TOP=51214.0; Wafer_64k=54429.0; Wafer_Slit=19794.0; Wafer_Count=1457.0; Wafer_32k=19794.0; CATHODE=54428.0;
							EVENT	15 Mar 2024 14:59	laurent burtin	Maintenance		SYMPTOM SELECTED: faute rf
							ANALYSIS	15 Mar 2024 15:00	laurent burtin	Maintenance		Vu sur APC, derive range depot rf side
							PASSDOWN	15 Mar 2024 15:03	laurent burtin	Maintenance		Selon Matthieu il faut changer la capa bank. Commande Capa Bank.
							PASSDOWN	15 Mar 2024 15:08	laurent burtin	Maintenance		Hv capa bank: Ref ST: 9REPE333 Ref AMAT:0190-18136
							PASSDOWN	15 Mar 2024 18:06	laurent burtin	Maintenance		La HV capa bank est sur la servante au basement.
							ACTION	15 Mar 2024 21:52	lionel santini	Maintenance		Avec Jérôme chgt HV CAPA BANK OLD 00406787AMI NEW 99554881 AMI
							PASSDOWN	15 Mar 2024 21:54	lionel santini	Maintenance		OLD HV CAPA BANK laissée sur servante Basement en Attente confirmation Diag
							ACTION	15 Mar 2024 22:22	lionel santini	Maintenance		Attente cyclage BOX19
							PASSDOWN	15 Mar 2024 22:23	lionel santini	Maintenance		envoi cyclage BOX19
							ACTION	16 Mar 2024 01:24	Jerome mouyon	Maintenance		APC à checker : Si APC OK Qual à faire post chgt HV CAPA BANK SIDE Si APC HS Diag à reprendre
							ACTION	16 Mar 2024 02:38	laurence car	Maintenance		apc OK QUAL
							EVENT	16 Mar 2024 02:39	laurence car	Qualification		qualif ok
EVENT	16 Mar 2024 02:40	laurence car	Qualification		MAINTENANCE FINISHED CAUSE RF-AUTRE CAUSE RF-BIAS MATCH							
EVENT	16 Mar 2024 02:42	laurence car	Qualification		CHECKLIST E192966 : STARTED AMAGCENTURA TPMC MANDATORY QUAL DGF BOX POST CM							
												CHECKLIST E192966 : VALIDATED AMAGCENTURA TPMC MANDATORY QUAL DGF BOX POST CM

Source: STMicroelectronics TPMC System

The figure illustrates various fields selected to enhance the prompt and enable the conversational assistant. The field *ANALYSIS*, highlighted in green, contains the technician’s initial acknowledgment. The *PASSDOWN* field, marked in yellow, includes natural language input for the handover of maintenance tasks when there is a shift change, ensuring continuity in the work order. Actions taken during the work order are recorded in the *ACTION* field, shown in darker blue. This field often contains a mix of languages, specialized maintenance terminology specific to the company, and varies

depending on the technician's detail in documenting the actions that led to problem resolution. The *EVENTS* field, in light blue, has been excluded from use; as per user feedback, it does not provide directly useful information.

For the longitudinal downtime file, a representative document is depicted in Figure 14.

Figure 14 – Example of a long down file.

DFG01B		
22-Sep-19	puissance staure à 1000 W	
	remplacé ssd side s/n 03171707em , par 99430598SA	SSD side
	faute 45 V low sur side et qq faute 60	
	Pris SSD Sur DGF03B (LOG) IN: 218320 OUT: 99430598SA 45 V ok mais faute 60 side	SSD side DGF03B
	Mesure du connecteur J6-6 et J6-7 sur la master, J6-6 HS a 150hms vs 300KOhms normalement	
a faire	a monter sur DGF01B: 1-carte master 194101 (ex master top DGF03C , symptôme se reset avec fte 08 sur side, mais tj meme pb apres son remplacement sur DGF03C) 2-SSD side 99430482SA (ex DGF09A, symptôme faute 60, mais en fait pb de capa C4)	
	master side DGF01B a reparer faute 60 en CC	
	150-18048 cable J1 sur deck side	

Source: STMicroelectronics long down file

As observed, the documentation within the long down files lacks a standardized structure and does not adhere to a consistent writing pattern. These documents are manually compiled by machine specialists and detail the actions taken to resolve specific issues. Although considered secondary in the problem analysis, these files can provide valuable insights when addressing persistent problems.

5.4 DATA ANALYSIS AND EXPLORATION

The examination of various data structures and sources played a key role in developing a data treatment strategy for the Large Language Model (LLM). The style and composition of the documents significantly influence the quality and precision of the generated content. Direct token substitution with domain-specific jargon often leads to a loss of meaning, affecting the accuracy of the responses.

A nuanced strategy was adopted to address these issues. The approach involves enriching the original natural language inputs with additional relevant data to preserve context and technical nuances, thereby enhancing the LLM's ability to generate accurate and meaningful answers. This enrichment process helps the model understand

the intricacies of the maintenance domain and provide contextually appropriate and technically insightful responses.

The strategy emphasizes the importance of a comprehensive understanding of the data and the domain to tailor the LLM effectively. It requires an iterative process of testing and refining data inputs and the model's interpretive algorithms to ensure the output meets the desired standards of relevance and technical accuracy. The aim is to create an LLM that can interact with users in a conversational manner, providing reliable information and support that is informed by the specificities of the maintenance field.

6 RESULTS AND DISCUSSION

The present session will expose the findings after the implementation of the proposed methodology and the metrics achieved while performing test sessions with the user, also, the modifications of what was expected in the projected management and what was done in practice.

6.1 TPMC MODULE

For the TPMC module, as discussed in the Materials and Methods section, the ingested documents originate from an Oracle database containing various details of work orders. The most utilized input was the technician's comment, which includes natural language descriptions of the actions taken and what specifically resolved the original issue. As anticipated, some limitations were encountered concerning the clarity of the information presented in the work orders. It was not always evident which action directly addressed the problem. However, in numerous cases, the developed system successfully extracted the pertinent information within the given context and provided insights that led to the resolution of a problem.

The initial tests with the *CSV* data and the implemented pipeline of the data ingestion engine, where documents were split and then stored in the database, indicated that this data structure was not optimal for the retrieval framework's use. Consequently, a more straightforward method was adopted to provide users with options to select various filter combinations. The subsequent retrieval of information could then augment the prompt with all work orders and their contents for the LLM.

The database features utilized were modified based on the alpha release feedback, resulting in a new set comprising: work order id, work order start datetime, type of maintenance (CM, PM, *Non Scheduled Maintenance* (NSM)), machine family, machine name, event type, comments, and event start datetime. By arranging this data chronologically according to the event datetime, it was possible to structure the context and various event types in a manner that highlighted the sequence of steps and ultimately identified the action that led to the problem's resolution. The filtered dataframe was then parsed into JSON format, as the LLM demonstrated improved performance with this data structure, and it offered a context format that was less costly in terms of token usage.

In response to the initial user interactions, it was deemed highly valuable to implement a search engine capable of querying specific terms within work orders and retrieving all relevant entries from various services containing the term. The search functionality was particularly applied to the pre-treated comment column because the token replacement step altered the sentence structure, thereby hindering the retrieval of necessary information. The partial ratio (fuzzywuzzy python library) technique was

employed to search for terms based on similarity, with scores categorized into four levels: low, medium, high, and exact, corresponding to score ranges of ([40,60,80,100]) respectively. The partial ratio technique operates by employing a string matching algorithm that calculates the Levenshtein distance to estimate the difference between sequences. This approach allows for the comparison of a search term to existing text within the database, even if the match is not exact, thus enhancing the robustness of the search engine in identifying relevant information.

The interaction of the user with the developed tool is of extreme importance and for the given reason a graphical interface was developed merging the interface of a usual chatbot in the style of the ChatGPT and also the filter structures present on the TPMC from STMicroelectronics in which the technician works in a daily basis. The Figure 15 illustrates the main page of the interface and the different options to interact with. The graphical interface was developed using the python streamlit library.

Figure 15 – Main page of the User Interface.

The screenshot shows the main page of the User Interface. The interface is divided into two main sections. On the left is the 'User Input' sidebar, which contains several interactive elements: a 'Parameters' section with the STMicroelectronics logo and 'life.augmented' text, a 'Reset Chat' button, a 'Use database' toggle switch (currently on), an 'Answer evaluation' dropdown menu set to '1 - False information', an 'Evaluate answer' button, a 'Base instruction' dropdown menu set to 'Choose an option', a 'Machine name' dropdown menu, and 'Start date' and 'End date' input fields. On the right is the main chat area, which displays the title 'STMMicroelectronics', the subtitle 'Augmented Maintenance', and a message: 'Context is too large, wo: 23778, modify the filter options'. At the bottom of the chat area is a text input field with the placeholder 'Say something' and a send button.

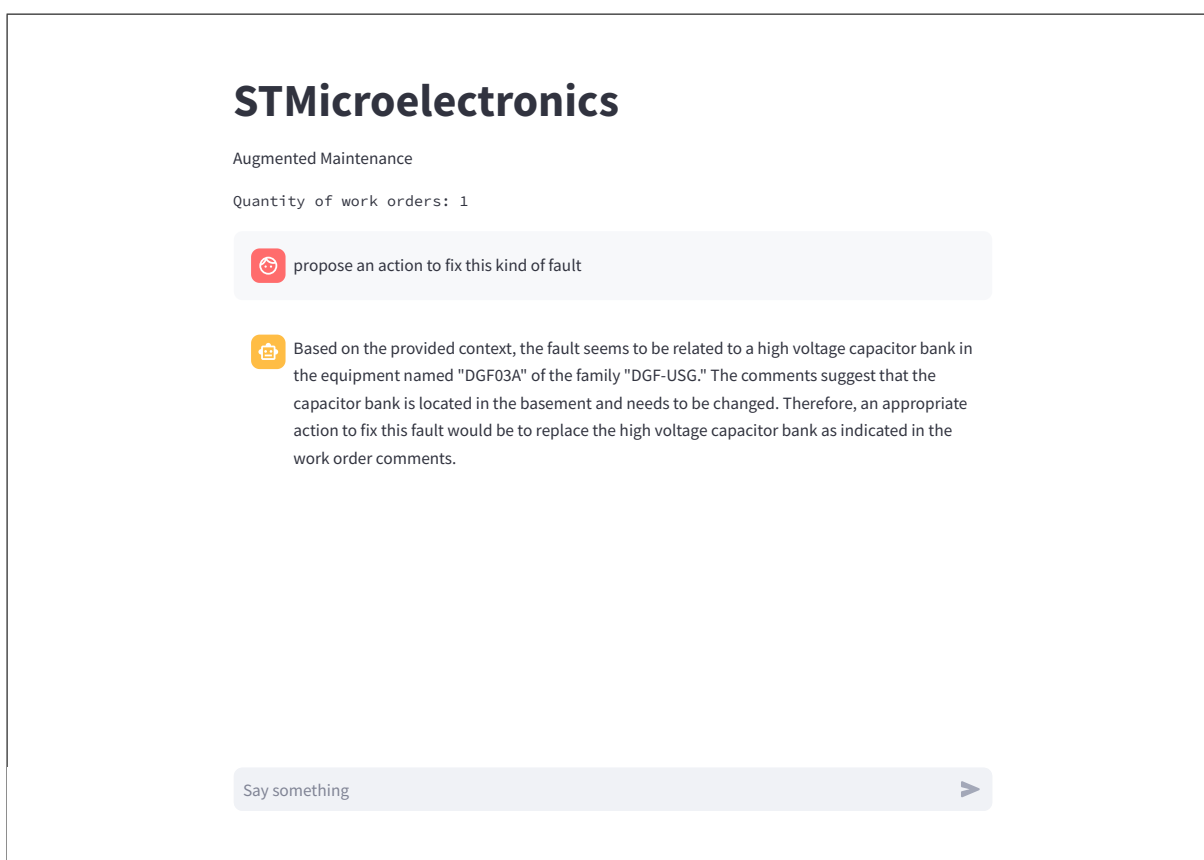
Source: Author

After opting to use the database, the system presents the user with various filters, some of which are illustrated in the figure, including options like base instruction selection and machine name. During the testing sessions with end-users—four sessions to date, each lasting one hour—additional base instructions were developed beyond the standard short and long summarization. These new instructions, "identify root cause"

and "list spare parts replacements," were created to address users' specific needs for locating pertinent information, which they previously had to do manually and could not accomplish through the term search feature in the TPMC system. Below the reset chat button, there is a select box for evaluating the system's response. This feature logs user input values, RAG system answer, time taken to generate the response, model used, base instruction, score, and timestamp.

An example of usage of the RAG system based on the work order presented in Figure 13 as context is presented in Figure 16.

Figure 16 – User interaction with the RAG system for one work order.



Source: Author

The base instruction selected to generate this response was of the standard type and, as validated by the user/expert, contains the correct information and provides a useful answer that directly addresses the issue by suggesting that an action to solve a problem similar to the one presented in the work order would be the replacement of the high voltage capacitor bank. The manner in which the information is presented and the language used are very important for its acceptance by the end user, who may or may not be familiar with the use of Generative AI. For this reason, it is essential to have an intuitive interface and proper documentation to guide the best practices to be followed in order to generate the appropriate responses. Another example of usage, now using


the summarizing task for the RAG system based on the work order presented in Figure 13 as context is presented in Figure 17.


Figure 17 – Summarization of example work order.

STMicroelectronics

Augmented Maintenance

Quantity of work orders: 1

 Summarize the work order



- **Type of Intervention:** CM (Corrective Maintenance)
- **Machine(s):** Equipment name: DGF03A, Family name: DGF-USG
- **Work Order ID:** 1064326372
- **Date:** The work order was initiated on 18-11-2004 17:25:14

Analysis: The maintenance intervention was initiated due to an observation of a vacuum (vu) on the APC (fault detection control) and the need to change the high voltage capacitor bank based on Matthieu's recommendation. The high voltage capacitor bank was located in the basement of the servante.

Actions:

1. Changed the old high voltage capacitor bank (00406787ami) with a brand new one (99554881ami).
2. Left the new high voltage capacitor bank on the servante in the basement, awaiting confirmation of the diagnostic results.
3. Sent a cycling command for box19.
4. Checked the fault detection control (APC). If the APC was okay and qualified, post change the high voltage capacitor bank side. If the APC was out of order, perform a diagnostic.
5. Confirmed that the APC was okay and qualified.
6. Performed the qualification, which was successful.

Conclusion: The high voltage capacitor bank was replaced with a new one. The new capacitor bank was left on the servante in the basement, awaiting confirmation of the diagnostic results. The fault detection control (APC) was checked and found to be okay and qualified.

Probable cause based on actions: The probable cause of the issue was a faulty high voltage capacitor bank, as indicated by the need to replace it.

Say something ➤

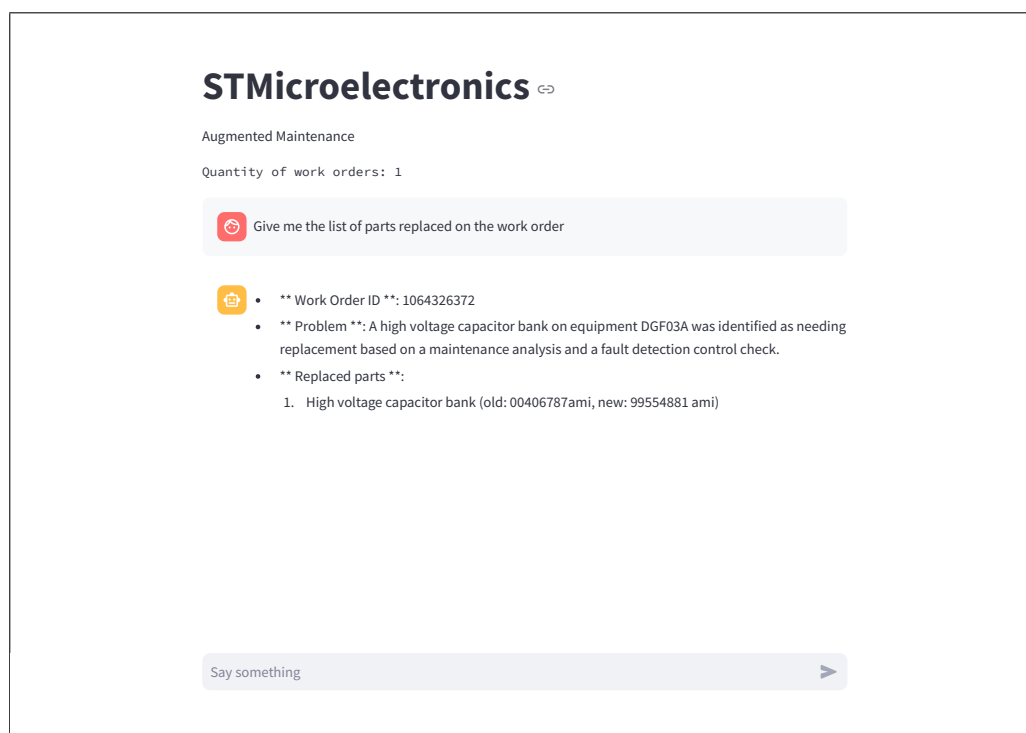
Source: Author

As shown in the figure above, by employing a base instruction of the 'short' type within the same context as the 'advise an action' task, the system was able to chronologically list the actions taken along with their respective outcomes. It even pinpointed

that the solution to the problem involved replacing the capacitor bank, demonstrating the repeatability of the results when generating responses based on context.

Another task proposed by the user and then developed was the part replacement base instruction and an example is presented in Figure 18.

Figure 18 – List of part replacements for work order 1064326372.



Source: Author

As previously mentioned, a base instruction was provided to the system to specify the type of information it should seek in the context of the work order. The example in Figure 18 illustrates a good behavior of the RAG agent, which was validated by specialists. Its response is direct, precise, and lists all the parts that were installed, providing useful information to the user and insights for solving similar problems. For the work order analyzed here, the technician clearly stated in one of the action fields that he replaced the high voltage capacitor bank of a certain part number with a new one, which resolved the problem. This is implicit in the sequence of actions and the pasdown shown in Figure 13, where after replacing the part, the technicians applied qualification to the machine and verified that it returned to operational status. This is also due to the data treatment step, especially the token replacement using the dictionary, as the comments do not follow a standard writing format and include words specific to the private context of STMicronelectronics maintenance and this factory.

However, the system does not always follow the base instructions correctly. In some cases, depending on the context and the work order, asking the same question with a different work order yields results that are not as trustworthy as the example


presented before. This problem can be related to various issues, such as context length, model capabilities and size, the quality of the context used, and the generation strategy for the output parameters of the LLM. An example of this problem is presented while performing the same task of part listing for another set of work orders, as illustrated in Figure 19.


Figure 19 – List of part replacements for work orders filtered by the term 'faute 92'.

STMicroelectronics

Augmented Maintenance

Quantity of work orders: 4

 List the parts replaced by work order


 Based on the provided context, the following parts were replaced:

- Work Order ID: 1064331043, Equipment Name: dgf06a
 - Replaced part 1: ssd (ssd board), Out: 179486, Fault: 92, In: 212696
 - Replaced part 2: pdb (Power Distribution Board), Fault: 92
 - Replaced part 3: connector 45v (45 volts) for ssd
- Work Order ID: 1064334247, Equipment Name: dgf04c, Family Name: dgf-usg
 - Replaced part: None identified in the context
- Work Order ID: 1080071453, Equipment Name: dgf02b
 - Replaced part 1: None identified in the context

Work Order ID: 1064329628, Equipment Name: dgf04c, Family Name: dgf-usg

- Replaced part: None identified in the context

The context does not provide enough information to determine if any parts were replaced for Work Order ID: 1064331043, Equipment Name: dgf06a (second occurrence).



Source: Author

The above presented example of a task that used the base instruction of the replaced part list consisted in the filtered context of work orders from the date range of first of March of 2024 to the present and using the search term with exact correspondence for the expression '*faute 92*' related to the fault number 92. While the first work order ('1064331043') have the replaced parts listed correctly, the other work orders ('1064334247', '1080071453' and '1064329628') don't and the message '*None identified in the context*' is displayed as this is false, due to the action listed in '1064329628': *Fait Changement SSD Side OLD SN: 02021044EM NEW SN: 02441532EM (PB de range RF) Test débit RF OK*, that means that the replacement of the SSD was performed, for '1080071453': *Avec Jeannot : Remplacement de la SSD TOP SN old SSD : 234 594 Envoie ocap en PSG04 Faute RF top au step 5 de heat up : sp 300W > le géné sature à 2000w Pas de faute physique sur le géné top* indicates the replacement of SSD TOP also.

As discussed before, this result can have many explanations and one of them that it is the most probable one is the context length being too large due to the quantity of data and also the quality of data. This hypothesis is confirmed by performing the task listing individually for each work, reducing the context length and providing more precision.

While this can improve the quality of the answer as the attention mechanism will focus on the important information for each work order, the time to generate and the vision of the whole problem could be lost, like the link between different work orders in different occasions that may have cause and consequence relations. Being more clear, the events of a work order in the past directly or indirectly impacted the persistent error encountered in a subsequent work order in the future.

However, more tests and evaluations must be put in place to determine if this kind of problem is very frequent as due to time constraints it was not possible to asses during the present work. Other options would be to improve the data treatment mechanism and test different base instruction compositions and the language used for generation and inferences. In order to evaluate the problem aforementioned and apply a solution, the context format of the work orders was modified and the organization of the information delivered to the LLM was adjusted. Additionally, only one work order at a time from the four found with the search term context was tested. As a result, the system was finally able to correctly list the parts that were replaced. This indicates that the problem lies in the context length and format, as well as the lack of resources to use models with larger input sizes. While testing the work orders individually, another problem related to input size (context length) and resources was found. The total available memory of the GPU used reached its maximum (80[GB]), preventing the generation specifically for the work order '1080071453,' which had an input size of 5200 and required 30[GB] of memory, but only 10[GB] were available after loading the model (LLM and embedding).


This issue can be addressed in different ways, such as batch inference and the use of multiple GPUs or GPUs with large available memory for local use.


Figure 20 – Root cause for work order 1064326372.

STMicroelectronics

Augmented Maintenance

Quantity of work orders: 1

 Give me the root cause

 Root Cause Analysis:

Based on the provided work order information, the issue with the DGF03A equipment appears to involve a high voltage capacitor bank that was identified as being out of order during a fault detection control check. The root cause hypothesis is that the high voltage capacitor bank was faulty and required replacement.

Evidence:


The work order contains several entries indicating that the high voltage capacitor bank was identified as being out of order during a fault detection control check (MAINT_EVENT_TYPE: "PASSDOWN", MAINT_EVENT_COMMENTS: "apc (fault detection control) hs (out of order) diag (diagnostic) reprendre"). Subsequently, a maintenance action was taken to change the high voltage capacitor bank (MAINT_EVENT_TYPE: "ACTION", MAINT_EVENT_COMMENTS: "avec jrome chgt (change) hv (high voltage) capa (capacitor) bank old 00406787ami new (brand new) 99554881 ami old hv (high voltage) capa (capacitor) bank laisse sur servante basement en attente confirmation diag (diagnostic)"). This evidence supports the hypothesis that the high voltage capacitor bank was the root cause of the issue.

Conclusion:

The root cause of the issue with the DGF03A equipment was a faulty high voltage capacitor bank, which was identified during a fault detection control check and subsequently replaced in a maintenance action.

Final Conclusion:

The maintenance intervention resulted in the replacement of the faulty high voltage capacitor bank on the DGF03A equipment. No further actions are required at this time.

Say something


Source: Author

In this example of the task of root cause analysis, it is possible to see the structure defined in the base instruction: create a hypothesis about the root cause, find evidence to sustain it, and finally create a conclusion to summarize and link the events

between what happened and led to the problem and finally what solved the problem, the answer from the System was validated by the specialist as complete and useful information. This kind of task allows the technician to interact with the data and understand better and more straightforwardly the historical information on past problems to solve persistent issues. Once again, the token replacement evidenced in Figure 20 as '*chgt (change) hv (high voltage) capa (capacitor) bank old*' plays a crucial role in reducing the chance of hallucinations in the generation. However, creating or implementing other methodologies yet to be explored and tested could develop even better results, suggesting that working with the original language of the data could be investigated and evaluated in comparison with the present results.

In summary, the results obtained with the proposed module and with the current resources yielded good results based on the user feedback and have space to development and improvement focusing on the industrial environment and needs. The module has demonstrated its value and potential as an alpha release and as a tool for exploration. With continued development and the integration of additional data sources and techniques, its performance could be further enhanced.

6.2 LONG DOWN MODULE

The proposal for the 'long down' module involved replicating the chatbot development process and then facilitating a discussion about the content of the files, as well as potential decisions and actions required to return the machine to an operational state. However, since the information contained in the files was the same as that in the TPMC, with the addition of some visual context manually input by the technician, it was determined that these should not be the main focus of the releases due to time constraints and other resource limitations.

In experimenting with the RAG (Retrieval-Augmented Generation) methodology, the system's ability to retrieve information based on the vector storage of document embeddings was tested and validated. The accuracy with which the system could identify document sections containing the required information was confirmed. Other techniques, such as knowledge graphs and fine-tuned embedding models, have shown promise for retrieving tasks with this type of unstructured data and should be further explored.

6.3 TOOL'S EVALUATION BASED ON USER'S FEEDBACK

As previously mentioned, the end users' availability is constrained due to their responsibilities at the manufacturing site. Nonetheless, the Rousset's maintenance team was able to participate in several discussions regarding the solution, its requirements, potential improvements, and, in the final development phase, they engaged in four test-

ing sessions to evaluate the tool. These sessions aimed to collect metrics to assess whether the RAG system was not only correct but also genuinely useful. The results of the four test sessions, along with the scores assigned, are presented in Table 5.

Table 5 – Collected metrics during test session.

Base Instruction	Average	Normalized Average	Average Generation Time [s]	Tests Count
summarize_short	2.64	76.40	23.8	11
summarize_long	2.33	73.30	60.92	9
standard	3.6	86.00	27.14	10
root_cause	3.5	85.00	30.24	8
part_replacement	4.71	97.10	19.89	7

Source: Author

Up to the end of the present work, 45 tests have been performed and evaluated by the user. From the metrics Table 2, we can infer that the system performs well in terms of the accuracy of facts based on the given context and the assumptions made for the machines, work orders, and other applied filters. In terms of macro metrics, regardless of the base instruction used, the system has an **average score**: 3.27, a **normalized average**: 82.67, and an **average generation time**: 32.50 seconds.

The metrics, as previously mentioned, are proposed to enhance trust in the system and also to facilitate the development and maintenance of the framework. They provide a straightforward indication of whether there has been a degradation in response quality over time, which would warrant further exploration and investigation.

However, the experiments alone do not capture the significance of the tool or the user experience and the fulfillment of needs that were specified earlier for a dynamic environment like maintenance. From the outset of the project, the goal has not been to replace human labor but to place the individual at the heart of development. For this reason, the end users have been involved in various project phases, contributing their knowledge, experience, and needs.

7 CONCLUSION

The project investigated the application of the Retrieval-Augmented Generation (RAG) methodology utilizing maintenance data from STMicroelectronics. The principal objective was to create a tool designed to aid in proposing solutions to address the causes of machine failures, particularly focusing on persistent downtimes.

The data explored in this project included maintenance logs from corrective, preventive, and unscheduled interventions. Technicians, during their actions and analyses, input comments in natural language, documenting the events that transpired throughout the service order. The quality of this data is paramount; hence, a significant portion of the project was dedicated to data extraction, processing, and comprehension. These processes encompassed not only the translation of text from French to English but also the substitution of tokens that signified jargon unique to the industrial maintenance domain or specific to the context of STMicroelectronics. Such jargon falls outside the parametric knowledge of the large-scale language models employed, rendering this preprocessing step critical to ensure the accuracy and applicability of the responses generated by the system.

A qualitative evaluation methodology was implemented to gather user feedback, which is instrumental in determining whether the context and format in which information was presented to the models were suitable. Such feedback is vital to ensure that the system's generated responses are not only useful and precise but also meet the practical requirements of maintenance technicians. Additionally, the development of this evaluation tool was motivated by the necessity to confirm that the information conveyed to the end user is correct and safe. In operations of this nature, inaccuracies can lead to more severe downtime situations, incurring higher costs and potentially having catastrophic outcomes.

The methodology proposed and the current work have demonstrated that the system holds potential, and with further exploration and improvements to its mechanisms, it could be deployed in a production environment, yielding significant benefits for the end user as indicated by the metrics and direct feedback received throughout the project. In parallel with this work, other projects have been testing different applications of the RAG methodology across various knowledge domains. Based on these interactions and a review of the literature, it is suggested that future work could investigate more advanced evaluation techniques, such as RAGAS, for implementation. To expand the solution's usage dimensions, infrastructure and resource requirements must also be considered to ensure availability for end users.

The work undertaken has been immensely valuable for my development as an engineer, enhancing my skills in the domains of management, interpersonal relationships, and technical expertise.

REFERENCES

ACHOUCH, Mounia; DIMITROVA, Mariya; ZIANE, Khaled; SATTARPANAH KARGANROUDI, Sasan; DHOUIB, Rizck; IBRAHIM, Hussein; ADDA, Mehdi. On Predictive Maintenance in Industry 4.0: Overview, Models, and Challenges. **Applied Sciences**, v. 12, n. 16, 2022. ISSN 2076-3417. DOI: 10.3390/app12168081. Available from: <https://www.mdpi.com/2076-3417/12/16/8081>.

AFZAL, Muhammad; LI, Rita Yi Man; SHOAIB, Muhammad; AYYUB, Muhammad Faisal; TAGLIABUE, Lavinia Chiara; BILAL, Muhammad; GHAFLOOR, Habiba; MANTA, Otilia. Delving into the Digital Twin Developments and Applications in the Construction Industry: A PRISMA Approach. **Sustainability**, v. 15, n. 23, 2023. ISSN 2071-1050. DOI: 10.3390/su152316436. Available from: <https://www.mdpi.com/2071-1050/15/23/16436>.

AI@META. Llama 3 Model Card, 2024. Available from: https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

ARENA, S.; FLORIAN, E.; ZENNARO, I.; ORRÙ, P.F.; SGARBOSSA, F. A novel decision support system for managing predictive maintenance strategies based on machine learning approaches. **Safety Science**, v. 146, p. 105529, 2022. ISSN 0925-7535. DOI: <https://doi.org/10.1016/j.ssci.2021.105529>. Available from: <https://www.sciencedirect.com/science/article/pii/S0925753521003726>.

ARENA, Simone; FLORIAN, Eleonora; ZENNARO, Ilenia; ORRÙ, Pier Francesco; SGARBOSSA, Fabio. A novel decision support system for managing predictive maintenance strategies based on machine learning approaches. **Safety science**, Elsevier, v. 146, p. 105529, 2022.

BARATA, João; KAYSER, Ina. Industry 5.0 – Past, Present, and Near Future. **Procedia Computer Science**, v. 219, p. 778–788, 2023. CENTERIS – International Conference on ENTERprise Information Systems / ProjMAN – International Conference on Project MANagement / HCist – International Conference on Health and Social Care Information Systems and Technologies 2022. ISSN 1877-0509. DOI: <https://doi.org/10.1016/j.procs.2023.01.351>. Available from: <https://www.sciencedirect.com/science/article/pii/S1877050923003605>.

BAS VAN OUDENHOVEN PHILIPPE VAN DE CALSEYDE, Rob Basten; DEMEROUTI, Evangelia. Predictive maintenance for industry 5.0: behavioural inquiries from a work system perspective. **International Journal of Production Research**, Taylor& Francis, v. 61, n. 22, p. 7846–7865, 2023. DOI: 10.1080/00207543.2022.2154403. eprint: <https://doi.org/10.1080/00207543.2022.2154403>. Available from: <https://doi.org/10.1080/00207543.2022.2154403>.

BELLETIER, Clément; CHARKHABI, Morteza; PIRES DE ANDRADE SILVA, Gustavo; AMETEPE, Kevin; LUTZ, Mathieu; IZAUTE, Marie. Wearable cognitive assistants in a factory setting: a critical review of a promising way of enhancing cognitive performance and well-being. **Cognition, Technology & Work**, Springer, v. 23, p. 103–116, 2021.

BHARDWAJ, Abhijeet Sandeep; VEERAMANI, Dharmaraj; ZHOU, Shiyu. Identifying equipment health status from maintenance records using Lexicon based Unsupervised Sentiment Analysis Adjusted for Negation (LUSAA-N). **Computers & Industrial Engineering**, Elsevier, v. 186, p. 109693, 2023.

BOUSDEKIS, Alexandros; APOSTOLOU, Dimitris; MENTZAS, Gregoris. Predictive maintenance in the 4th industrial revolution: Benefits, business opportunities, and managerial implications. **IEEE engineering management review**, IEEE, v. 48, n. 1, p. 57–62, 2019.

BROWN, Tom et al. Language models are few-shot learners. **Advances in neural information processing systems**, v. 33, p. 1877–1901, 2020.

BUTT, Javid. A strategic roadmap for the manufacturing industry to implement industry 4.0. **Designs**, MDPI, v. 4, n. 2, p. 11, 2020.

CHIEN, Chen-Fu; CHEN, Chia-Cheng. Data-driven framework for tool health monitoring and maintenance strategy for smart manufacturing. **IEEE Transactions on Semiconductor Manufacturing**, IEEE, v. 33, n. 4, p. 644–652, 2020.

CHOMKLIN, Amonpan; JAIYEN, Saichon; WATTANAKITRUNGROJ, Niwan. A Survey of AI Techniques based on Predictive Maintenance in Lean Manufacturing. **Science, Technology, and Social Sciences Procedia**, v. 2023, n. 4, p. cim03–cim03, 2023.

DOANH, Doung Cong; DUFEK, Zdenek; EJDYS, Joanna; GINEVIČIUS, Romualdas; KORZYNSKI, Pawel; MAZUREK, Grzegorz; PALISZKIEWICZ, Joanna;

- WACH, Krzysztof; ZIEMBA, Ewa. Generative AI in the manufacturing process: theoretical considerations. **Engineering Management in Production and Services**, v. 15, n. 4, p. 76–89, 2023.
- ES, Shahul; JAMES, Jithin; ESPINOSA-ANKE, Luis; SCHOCKAERT, Steven. **RAGAS: Automated Evaluation of Retrieval Augmented Generation**. [S.l.: s.n.], 2023. arXiv: 2309.15217 [cs.CL].
- EUROPÉENNE, Commission;
L'INNOVATION, Direction générale de la recherche et de; BREQUE, M; DE NUL, L;
PETRIDIS, A. **Industrie 5.0 : vers une industrie européenne durable, centrée sur l'humain et résiliente**. [S.l.]: Office des publications de l'Union européenne, 2023.
DOI: doi/10.2777/2866.
- EZUKWOKE, Kenneth; TOUBAKH, Houari; HOAYEK, Anis;
BATTON-HUBERT, Mireille; BOUCHER, Xavier; GOUNET, Pascal. Intelligent fault analysis decision flow in semiconductor industry 4.0 using natural language processing with deep clustering. In: IEEE. 2021 IEEE 17th international conference on automation science and engineering (CASE). [S.l.: s.n.], 2021. P. 429–436.
- FLORIAN, E; SGARBOSSA, F; ZENNARO, I. Machine learning for predictive maintenance: A methodological framework. **Proceedings of the XXIV Summer School “Francesco Turco”—Industrial Systems Engineering, Bergamo, Italy**, p. 9–11, 2020.
- FLORIAN, Eleonora; SGARBOSSA, Fabio; ZENNARO, Ilenia. Machine learning-based predictive maintenance: A cost-oriented model for implementation. **International Journal of Production Economics**, Elsevier, v. 236, p. 108114, 2021.
- FREIRE, Samuel Kernan; WANG, Chaofan; FOOSHERIAN, Mina;
WELLSANDT, Stefan; RUIZ-ARENAS, Santiago; NIFORATOS, Evangelos. Knowledge Sharing in Manufacturing using Large Language Models: User Evaluation and Model Benchmarking. **arXiv preprint arXiv:2401.05200**, 2024.
- GAO, Yunfan et al. **Retrieval-Augmented Generation for Large Language Models: A Survey**. [S.l.: s.n.], 2024. arXiv: 2312.10997 [cs.CL].
- GOTO, DAISUKE; MIKAMI, NAOTAKA; YAMAUCHI, TAKAHIRO;
NAGATA, YOSHIYUKI; HATTORI, MIRI; MAKOTO, TSUTSUI. Smart Maintenance and

Remote Monitoring by TOMONI® Utilizing Generative AI: Current Status and Future Prospects. **Mitsubishi Heavy Industries Technical Review**, v. 60, n. 4, p. 1, 2023.

GU, Zhaopeng; ZHU, Bingke; ZHU, Guibo; CHEN, Yingying; TANG, Ming; WANG, Jinqiao. Anomalygpt: Detecting industrial anomalies using large vision-language models. In: 3. PROCEEDINGS of the AAAI Conference on Artificial Intelligence. [S.l.: s.n.], 2024. P. 1932–1940.

GUU, Kelvin; LEE, Kenton; TUNG, Zora; PASUPAT, Panupong; CHANG, Ming-Wei. **REALM: Retrieval-Augmented Language Model Pre-Training**. [S.l.: s.n.], 2020. arXiv: 2002.08909 [cs.CL].

IUNG, Benoit; LEVRAT, Eric. Advanced maintenance services for promoting sustainability. **Procedia CirP**, Elsevier, v. 22, p. 15–22, 2014.

JAVAID, Mohd; HALEEM, Abid; SINGH, Ravi Pratap. A study on ChatGPT for Industry 4.0: Background, potentials, challenges, and eventualities. **Journal of Economy and Technology**, Elsevier, v. 1, p. 127–143, 2023.

JEONG, Cheonsu. A Study on the Implementation of Generative AI Services Using an Enterprise Data-Based LLM Application Architecture. **Advances in Artificial Intelligence and Machine Learning**, Advances in Artificial Intelligence and Machine Learning, v. 03, n. 04, p. 1588–1618, 2023. ISSN 2582-9793. DOI: 10.54364/aauml.2023.1191. Available from: <http://dx.doi.org/10.54364/AAIML.2023.1191>.

JIANG, Albert Q. et al. **Mistral 7B**. [S.l.: s.n.], 2023. arXiv: 2310.06825 [cs.CL].

JIANG, Albert Q. et al. **Mixtral of Experts**. [S.l.: s.n.], 2024. arXiv: 2401.04088 [cs.LG].

KELEKO, Aurelien; KAMSU-FOGUEM, Bernard; HOUE NGOUNA, Raymond; TONGNE, Amèvi. Artificial intelligence and real-time predictive maintenance in industry 4.0: a bibliometric analysis. **AI and Ethics**, v. 2, Mar. 2022. DOI: 10.1007/s43681-021-00132-6.

KELEKO, Aurelien Teguede; KAMSU-FOGUEM, Bernard; NGOUNA, Raymond Houe; TONGNE, Amèvi. Artificial intelligence and real-time predictive maintenance in industry 4.0: a bibliometric analysis. **AI and Ethics**, Springer, v. 2, n. 4, p. 553–577, 2022.

LANGCHAIN, INC. **Q&A with RAG**. Acesso em: 06 maio 2024.

LEMES, L. C.; HVAM, L. **Maintenance Costs in the Process Industry: A Literature Review**. [S.l.: s.n.], 2019. P. 1481–1485. DOI: 10.1109/IEEM44572.2019.8978559.

LEWIS, Mike; LIU, Yinhan; GOYAL, Naman; GHAZVININEJAD, Marjan; MOHAMED, Abdelrahman; LEVY, Omer; STOYANOV, Ves; ZETTLEMOYER, Luke. **BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension**. [S.l.: s.n.], 2019. arXiv: 1910.13461 [cs.CL].

LEWIS, Patrick et al. **Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks**. [S.l.: s.n.], 2021. arXiv: 2005.11401 [cs.CL].

MA, Xinyin; FANG, Gongfan; WANG, Xinchao. LLM-Pruner: On the Structural Pruning of Large Language Models. In: OH, A.; NAUMANN, T.; GLOBERSON, A.; SAENKO, K.; HARDT, M.; LEVINE, S. (Eds.). **Advances in Neural Information Processing Systems**. [S.l.]: Curran Associates, Inc., 2023. P. 21702–21720. Available from: https://proceedings.neurips.cc/paper_files/paper/2023/file/44956951349095f74492a5471128a7e0-Paper-Conference.pdf.

MANTRAVADI, Soujanya; JANSSON, Andreas Dyrøy; MØLLER, Charles. User-friendly mes interfaces: Recommendations for an ai-based chatbot assistance in industry 4.0 shop floors. In: SPRINGER. **ASIAN Conference on Intelligent Information and Database Systems**. [S.l.: s.n.], 2020. P. 189–201.

MATYAS, Kurt; NEMETH, Tanja; KOVACS, Klaudia; GLAWAR, Robert. A procedural approach for realizing prescriptive maintenance planning in manufacturing industries. **CIRP Annals**, v. 66, n. 1, p. 461–464, 2017. ISSN 0007-8506. DOI: <https://doi.org/10.1016/j.cirp.2017.04.007>. Available from: <https://www.sciencedirect.com/science/article/pii/S0007850617300070>.

MIKOLOV, Tomas; SUTSKEVER, Ilya; CHEN, Kai; CORRADO, Greg S; DEAN, Jeff. Distributed representations of words and phrases and their compositionality. **Advances in neural information processing systems**, v. 26, 2013.

MOBLEY, R Keith. **Maintenance fundamentals**. [S.l.]: Elsevier, 2011.

MOYNE, James; ISKANDAR, Jimmy. Big Data Analytics for Smart Manufacturing: Case Studies in Semiconductor Manufacturing. **Processes**, v. 5, n. 3, 2017a. ISSN 2227-9717. DOI: 10.3390/pr5030039. Available from: <https://www.mdpi.com/2227-9717/5/3/39>.

MOYNE, James; ISKANDAR, Jimmy. Big data analytics for smart manufacturing: Case studies in semiconductor manufacturing. **Processes**, MDPI, v. 5, n. 3, p. 39, 2017b.

NARDO, Mario di; MADONNA, Marianna; ADDONIZIO, P.; GALLAB, Maryam. A Mapping Analysis of Maintenance in Industry 4.0. **Journal of Applied Research and Technology**, v. 19, p. 653–675, Dec. 2021. DOI: 10.22201/icat.24486736e.2021.19.6.1460.

PEREIRA, Rodrigo; LIMA, Claudio; PINTO, Tiago; REIS, Arsénio. Virtual Assistants in Industry 4.0: A Systematic Literature Review. **Electronics**, MDPI, v. 12, n. 19, p. 4096, 2023.

POÓR, P.; BASL, J.; ZENISEK, D. **Predictive Maintenance 4.0 as next evolution step in industrial maintenance development**. [S.l.: s.n.], 2019. P. 245–253. DOI: 10.23919/SCSE.2019.8842659.

PYTORCH FOUNDATION. **Language Modeling with nn.Transformer and torchtext**. Acesso em: 06 maio 2024.

RABELO, Ricardo J; ROMERO, David; ZAMBIASI, Saulo Popov. Softbots supporting the operator 4.0 at smart factory environments. In: SPRINGER. **ADVANCES in Production Management Systems. Smart Manufacturing for Industry 4.0: IFIP WG 5.7 International Conference, APMS 2018, Seoul, Korea, August 26-30, 2018, Proceedings, Part II**. [S.l.: s.n.], 2018. P. 456–464.

REISCHAUER, Georg. Industry 4.0 as policy-driven discourse to institutionalize innovation systems in manufacturing. **Technological Forecasting and Social Change**, v. 132, p. 26–33, 2018. ISSN 0040-1625. DOI: <https://doi.org/10.1016/j.techfore.2018.02.012>. Available from: <https://www.sciencedirect.com/science/article/pii/S0040162517316657>.

SIMEU-ABAZI, Zineb; SASSINE, Chadi. Maintenance integration in manufacturing systems: from the modeling tool to evaluation. **International Journal of Flexible Manufacturing Systems**, Springer, v. 13, n. 3, p. 267–285, 2001.

SMITH, David J. **Reliability, maintainability and risk: practical methods for engineers**. [S.l.]: Butterworth-Heinemann, 2021.

TAIWO, Ridwan; BELLO, Idris Temitope; ABDULAI, Sulemana Fatoama; YUSSIF, Abdul-Mugis; SALAMI, Babatunde Abiodun; SAKA, Abdullahi; ZAYED, Tarek. Generative AI in the Construction Industry: A State-of-the-art Analysis. **arXiv preprint arXiv:2402.09939**, 2024.

VAN OUDENHOVEN, Bas; VAN DE CALSEYDE, Philippe; BASTEN, Rob; DEMEROUTI, Evangelia. Predictive maintenance for industry 5.0: Behavioural inquiries from a work system perspective. **International Journal of Production Research**, Taylor & Francis, v. 61, n. 22, p. 7846–7865, 2023.

VASWANI, Ashish; SHAZEER, Noam; PARMAR, Niki; USZKOREIT, Jakob; JONES, Llion; GOMEZ, Aidan N; KAISER, Łukasz; POLOSUKHIN, Illia. Attention is all you need. **Advances in neural information processing systems**, v. 30, 2017.

VISHNU, C.R.; REGIKUMAR, V. Reliability Based Maintenance Strategy Selection in Process Plants: A Case Study. **Procedia Technology**, v. 25, p. 1080–1087, 2016. 1st Global Colloquium on Recent Advancements and Effectual Researches in Engineering, Science and Technology - RAEREST 2016 on April 22nd & 23rd April 2016. ISSN 2212-0173. DOI: <https://doi.org/10.1016/j.protcy.2016.08.211>. Available from: <https://www.sciencedirect.com/science/article/pii/S2212017316305655>.

WANG, Liang; YANG, Nan; HUANG, Xiaolong; YANG, Linjun; MAJUMDER, Rangan; WEI, Furu. Multilingual E5 Text Embeddings: A Technical Report. **arXiv preprint arXiv:2402.05672**, 2024.

XU, Xun; LU, Yuqian; VOGEL-HEUSER, Birgit; WANG, Lihui. Industry 4.0 and Industry 5.0—Inception, conception and perception. **Journal of Manufacturing Systems**, v. 61, p. 530–535, 2021. ISSN 0278-6125. DOI: <https://doi.org/10.1016/j.jmsy.2021.10.006>. Available from: <https://www.sciencedirect.com/science/article/pii/S0278612521002119>.

YAN, Jihong; MENG, Yue; LU, Lei; LI, Lin. Industrial Big Data in an Industry 4.0 Environment: Challenges, Schemes, and Applications for Predictive Maintenance. **IEEE Access**, v. 5, p. 23484–23491, 2017. DOI: 10.1109/ACCESS.2017.2765544.

YUNIAWAN, Dani; ITO, Teruaki; MOHAMAD, Effendi. **A Proposal of Total Productive Maintenance Training System**. [S.l.: s.n.], Nov. 2013.

ZENG, Zhiyuan; YU, Jiatong; GAO, Tianyu; MENG, Yu; GOYAL, Tanya; CHEN, Danqi. **Evaluating Large Language Models at Evaluating Instruction Following**. [S.l.: s.n.], 2023. arXiv: 2310.07641 [cs.CL].

ZHANG, Weiting; YANG, Dong; WANG, Hongchao. Data-Driven Methods for Predictive Maintenance of Industrial Equipment: A Survey. **IEEE Systems Journal**, v. 13, n. 3, p. 2213–2227, 2019a. DOI: 10.1109/JSYST.2019.2905565.

ZHANG, Weiting; YANG, Dong; WANG, Hongchao. Data-driven methods for predictive maintenance of industrial equipment: A survey. **IEEE systems journal**, IEEE, v. 13, n. 3, p. 2213–2227, 2019b.