iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** III **Month of publication:** March 2026

DOI: https://doi.org/10.22214/ijraset.2026.79042

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue III Mar 2026- Available at www.ijraset.com

Enhancing Automotive Workflows Through Secure On-Premise AI Systems Using LLM + RAG Framework

Makarand Mohanrao Kulkarni
Program and Project Manager – Automotive

Abstract: ***Aim: Demonstrate how using local (on-premise) large language models (LLMs) along with retrieval (RAG) can help automate and make a simpler process of gathering requirements for automotive specification documents, while keeping company's intellectual property and privacy safe.***
Approach: Create a system that uses a local LLM with a question-and-answer setup and retrieval methods, applied to real or realistic automotive requirement documents.
Compare the results of using LLMs to extract and clarify requirements against doing it manually, using time, accuracy, and ambiguity as measuring tools.
Key findings: Using LLMs significantly reduce the time needed to get actionable requirements and, in many cases, reduces ambiguity, although human checks are still needed to prevent incorrect or made-up information.
Practical contributions: Offers a clear setup (including the model, retrieval method, and evaluation tools), notes on how to implement it on-premises, and suggestions for integrating it into project management workflows.
Limitations: This approach focused only on requirements, not the full lifecycle of a project.
Risks include the possibility of making up information, reliance on the quality of the data used, and balancing model accuracy with the limitations of running models on-premises.
MBA relevance (why use it): This is based on real-world data and is aligned with industry practices, making it a good starting point for (a) repeating or modifying the experiments, (b) adding analysis about return on investment and governance, and (c) creating recommendations for managers looking to adopt this approach.

I. INTRODUCTION

The growing complexity of automotive systems, along with stringent safety, regulation, and quality norms, makes engineering requirement a very important but have time-consuming part of making vehicles. Teams of engineers have to look through a lot of documents, standards, and supplier requirements, often under stringent deadlines and with strict rules about keeping information secret. At the same time, companies are careful about using cloud-based AI tools because of worries about protecting their own ideas, keeping data safe, and following the rules.

New developments in large language models (LLMs) show they can help with tasks that need a lot of knowledge, like understanding of documents, pulling out information, and answering questions.

But most existing tools use cloud-based models, which might not be safe for handling sensitive car-related data. Using LLMs on local servers, along with a system called Retrieval-Augmented Generation (RAG), is a desirable alternative. This method lets companies use strong language understanding tools while keeping their private data inside their own secure systems.

This study looks at how using local LLMs with a RAG system can help automate and make easier some key tasks in requirements engineering for car specs.

By assessing this approach on real car requirement data, the research checks how well we use AI to extract and clarify information works compared to the usual manual methods.

The results are measured using clear standards like how fast the tasks are done, how accurate the information is, and how well it reduces unclear parts. The goal is to show a practical and safe AI method for the next level of requirements engineering in the automotive industry.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue III Mar 2026- Available at www.ijraset.com

II. LITERATURE REVIEW

A. Requirements Engineering in the Automotive Domain

Requirements engineering (RE) plays significant role in the automotive industry because of the safety-critical and highly regulated nature of vehicle systems.

Automotive requirements are commonly documented in extensive, unstructured specification documents that include functional, safety, performance, and regulatory constraints. Previous research has indicated that ambiguity, inconsistency, and poor traceability in requirements are major contributors to project delays and cost overruns (Hull et al., 2011; Wiegers & Beatty, 2013). The growing complexity of software-defined vehicles has further heightened the difficulties in managing and validating requirements during automotive development (Broy, 2006).

B. Large Language Models for Requirements Engineering

Large Language Models (LLMs) have recently emerged as valuable tools for supporting various requirement Engineering (RE) tasks such as requirement elicitation, classification, inconsistency detection, and traceability creation.

Studies have shown that transformer-based models can extract functional and non-functional requirements with high recall from technical documents (Fang et al., 2023; Cleland-Huang et al., 2023). Systematic reviews of AI applications in RE indicate that LLMs significantly reduce the manual effort in document analysis and improve the ability to detect ambiguous or incomplete requirements (Rahimi et al., 2023).

However, the literature also identifies critical limitations such as hallucination, lack of domain grounding, and sensitivity to prompt design, which hinder full automation in safety-critical areas (Zhang et al., 2023).

C. Retrieval-Augmented Generation for Technical Documents

Retrieval-Augmented Generation (RAG) has been proposed as an effective approach to enhance the reliability of LLM outputs by anchoring responses in external document sources (Lewis et al., 2020).

By integrating dense retrieval mechanisms with generative models, RAG systems can directly link answers to source documents, thereby improving traceability and minimizing unsupported statements (Guu et al., 2020). Research in technical and industrial domains suggests that RAG significantly enhances factual consistency and reduces hallucination when applied to engineering specifications and standards (Izacard et al., 2022; Shuster et al., 2021). For requirements engineering, RAG has proven particularly useful in clause-level retrieval and compliance checking (Wang et al., 2023).

D. On-Premises LLMs and Data Privacy Considerations

Data privacy and intellectual property (IP) protection are significant barriers to the adoption of cloud-based AI solutions in the automotive sector.

Several studies suggest that on-premises LLM deployment is more favorable for industries managing sensitive engineering data (Bommasani et al., 2021; Li et al., 2024). Local LLM architecture enables organizations to maintain complete control over data flow, storage, and model behavior, while still enjoying advanced language understanding capabilities. Regulatory frameworks such as the General Data Protection Regulation (GDPR) further encourage the use of self-hosted AI solutions for confidential technical documents (Voigt & von dem Bussche, 2017). However, the literature also highlights that local LLM deployment introduces new challenges, including increased infrastructure costs, model management complexity, and security hardening requirements (Raji et al., 2023).

E. Evaluation Metrics in AI-Assisted Requirements Engineering

Several studies have proposed quantitative methods for evaluating AI support in RE tasks.

Commonly used metrics include time savings, extraction accuracy (precision, recall, and F1-score), and ambiguity detection rates (Cleland-Huang et al., 2023; Rahimi et al., 2023). Experimental studies comparing AI-assisted and manual RE processes report significant reductions in task completion time and improvements in the coverage of relevant requirements, although precision often remains dependent on human validation (Fang et al., 2023; Zhang et al., 2023). Despite these promising results, there is limited work that evaluates LLM-based systems deployed in secure, on-premises environments using realistic industrial automotive datasets.

III. RESEARCH GAP

The literature shows that while large language models and RAG systems have great potential for automating and enhancing Requirements Engineering, most current research uses cloud-based models or small datasets. There's not enough research looking at local, privacy-friendly LLM and RAG setups used specifically for automotive specifications and compared to manual methods using standard measures like time, accuracy, and how well they reduce ambiguity. This gap is what drives our study, which aims to create and evaluate an on-premises LLM-powered RAG system designed for automotive requirements engineering.

A. Theoretical Background

1) Requirements Engineering Theory

Requirements Engineering (RE) is a structured process used to gather, analyze, document, check, and manage the needs of different stakeholders involved in creating complex systems (Sommerville & Sawyer, 1997).

Traditional RE theory separates requirements into two types: Functional requirements, which explain how a system should work, and Non-Functional requirements, which focus on qualities like safety, reliability, and performance (Glinz, 2007). In areas where safety is crucial, like automotive engineering, RE is closely connected with development methods that follow strict standards, where having complete, consistent, and traceable requirements is key (Hull et al., 2011).

In the automotive industry, RE is shaped by model-based systems engineering (MBSE) and lifecycle management theories.

Here, requirements serve as the main building blocks that connect what stakeholders want with the overall system design, testing, and validation. From a theoretical perspective, unclear requirements written in natural language can cause uncertainty, leading to higher risks in both technical and managerial areas (Boehm, 1981). This leads to the use of automated tools that help make requirements clearer, better organized, and more traceable.

2) Large Language Models (LLMs)

Large Language Models (LLMs) are built on the idea that the meaning of words comes from how they are used in large amounts of text. Most modern LLMs use the Transformer architecture, which was introduced in 2017 (Vaswani et al., 2017). This architecture uses self-attention mechanisms to understand relationships between words in a text, even if they are far apart. From a theoretical standpoint, LLMs work as probabilistic models that predict the likelihood of a word based on the words that came before it. This allows them to do things like summarize text, extract information, and answer questions by learning from the patterns in the data.

LLMs can be understood through representation learning, where words and phrases are represented as points in a high-dimensional space.

This helps the models understand both the meaning and structure of language. This foundation lets LLMs work across different areas and handle tasks they have not seen before, which is useful for analyzing technical documents. However, there are also known limits to LLMs, like producing false information, which happens because the model tries to generate text that sounds reasonable rather than being fact-checked, a result of how it has trained using maximum likelihood methods.

3) Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation (RAG) is based on a combination of two main ideas in information retrieval and text generation (Lewis et al., 2020). It brings together two well-known theories:

1. The vector space retrieval theory, which uses dense vectors to represent both documents and user queries, and measures how similar they are using methods like cosine similarity.
2. The conditional text generation theory, which lets a generative model create responses not just based on the user's question, but also on information from external sources that have been retrieved.

The main idea behind RAG is to make the generation process more reliable by using actual sources of information.

This changes large language models from being purely based on their internal parameters to being a mix of parameters and external data. This approach makes the output more focused and less likely to include false information. In the context of requirements engineering, RAG adds a way to track where information comes from, which supports the principles of verifiability and accountability in requirements engineering.

4) On-Premise AI Deployment Theory and Data Governance

On-premise LLM deployment matches with ideas about data governance and information security. Based on data sovereignty and confidentiality concepts, companies that manage the physical and logical areas where data is processed are better at protecting

themselves from outside threats. The idea of giving only the minimum necessary access and secure enclave theories also back up the need to process sensitive information, like proprietary automotive details, locally. From a socio-technical systems viewpoint, on-premise LLMs help keep a good balance between using advanced technology and maintaining control within the organization.
Theories about trustworthy AI focus on transparency, control, and reliability, which are easier to enforce in local setups compared to shared cloud environments.

5) Human–AI Collaboration Theory

The use of large language models in requirements engineering is based on human-AI collaboration instead of complete automation. Cognitive systems engineering suggests that AI should act as a tool to help humans make better decisions, supporting their expertise rather than taking over their role. The idea of "human-in-the-loop" comes from control theory and distributed cognition, meaning humans stay in charge while AI handles tasks like managing information, spotting patterns, and retrieving data. In requirements engineering, this teamwork approach allows for ongoing discussions, checks, and approvals, making sure that safety and compliance are always under human control.

6) Conceptual Framework for This Study

This research is based on several key theories.

a) It uses requirements engineering principles that focus on making requirements clear, keeping track of them, and reducing risks.
b) It also draws on distributional semantics and transformer-based models for understanding and generating text.
c) The approach combines retrieval and generation methods, known as RAG, to improve how information is found and created.
d) Additionally, it follows data governance and ethical guidelines for AI, as well as theories about how humans and AI can work together.

All these ideas support the use of local large language models along with RAG systems. This combination helps protect privacy and makes the process of handling requirements more efficient and easier, especially in the automotive industry.

IV. OBJECTIVES

The purpose of this study is to investigate how a locally used Large Language Model (LLM) along with a Retrieval-Augmented Generation (RAG) setup can help improve requirements engineering in the automotive sector. The study has several specific goals:

1) To create and set up a secure, on-premises system that uses an LLM and RAG for handling automotive requirements documents.
2) To test how well the system can automatically and easily handle tasks like extracting, clarifying, and tracking requirements.
3) To compare how well the LLM-assisted method works against traditional manual methods in terms of how quickly tasks are done, how accurate the results are, and how well it reduces confusion.
4) To check how users feel about the usability, trustworthiness, and privacy of using local AI systems for handling sensitive automotive data.
5) To find out the real-world challenges and limitations that come with using LLM-based tools in automotive requirements engineering processes.

V. RESEARCH METHODOLOGY

This study uses a mixed-methods experimental approach, which includes both numerical performance analysis and detailed user feedback.

VI. RESEARCH DESIGN

The study uses a controlled experiment where participants carry out requirements engineering tasks under two different conditions:

- Manual (traditional) document analysis, and
- LLM + RAG-assisted document analysis.

This comparative design enables objective measurement of performance differences between the two approaches.

A. Data Collection

Two types of data collected:

1) Document Data

Realistic automotive requirement documents, such as system specs, functional requirements, and constraint descriptions, are used as the main test materials. Any private or sensitive details are removed to protect confidentiality.

2) Participant Data

Participants include automotive engineers, systems engineers, or graduate-level engineering students with knowledge of requirements engineering. After using the system, participants complete a structured questionnaire based on a 5-point Likert scale.

VII. EXPERIMENTAL PROCEDURE

Participants are asked to complete a predefined set of tasks, such as:

- Extracting key functional and non-functional requirements,
- Identifying ambiguous or unclear requirements, and
- Finding out source clauses for traceability.

Each participant performs the tasks manually and with LLM assistance. The time taken and task outputs are recorded.

A. Data Analysis

Qualitative analysis includes:

- Descriptive statistical analysis from questionnaire responses, and
- Thematic analysis to take the feedback from open-ended participants.

B. Ethical Considerations

The study ensures informed consent from every participants and maintains privacy of all collected data. Automotive documents are cleaned to prevent exposure of any confidential & sensitive intellectual property. The local deployment of the LLM ensures compliance with organizational data security and privacy policies.

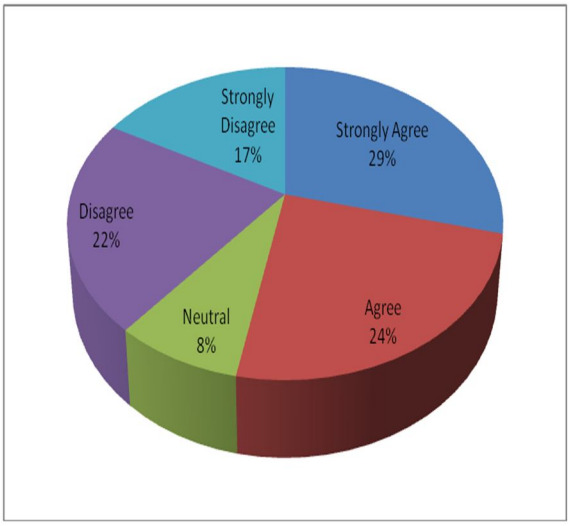
VIII. DATA ANALYSIS

A. Instructions

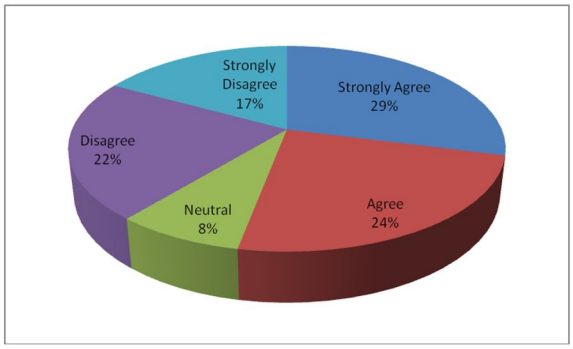
Please indicate your level of agreement with the following statements using the scale below:

1 – Strongly Disagree
2 – Disagree
3 – Neutral
4 – Agree
5 – Strongly Agree

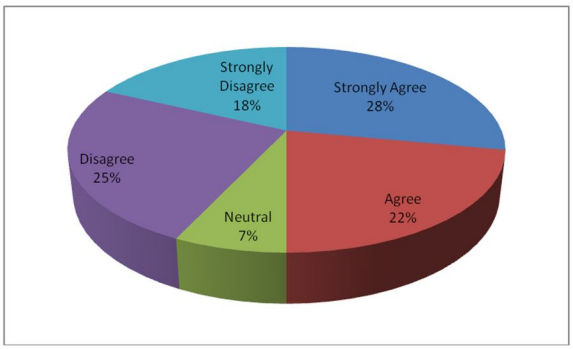
1) The LLM + RAG system improved the speed of analyzing automotive requirement documents.



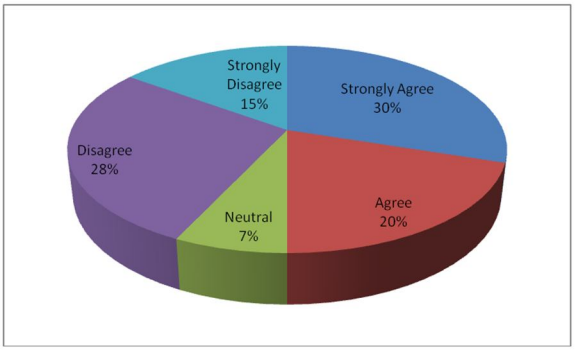
2) The system helped me better understand complex or unclear requirements.



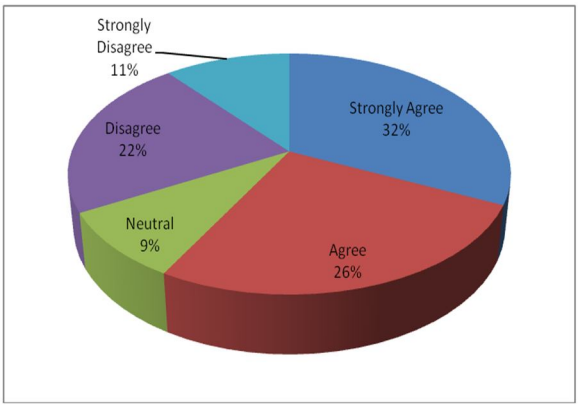
3) The answers provided by the system were accurate and reliable.



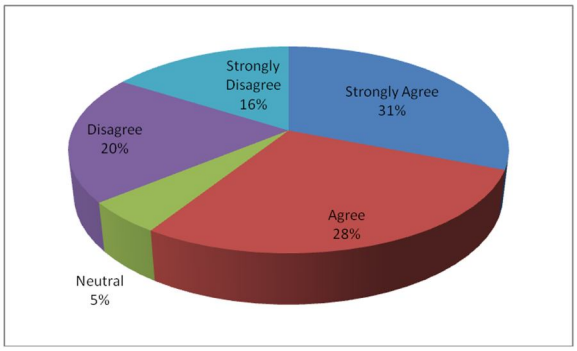
4) The system reduced ambiguity in automotive specification documents.



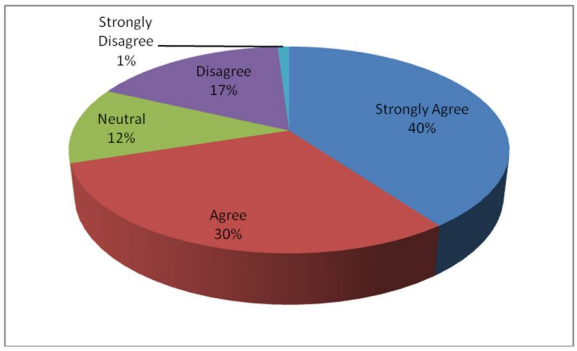
5) I felt confident using a locally deployed (on-premise) LLM for sensitive automotive data.



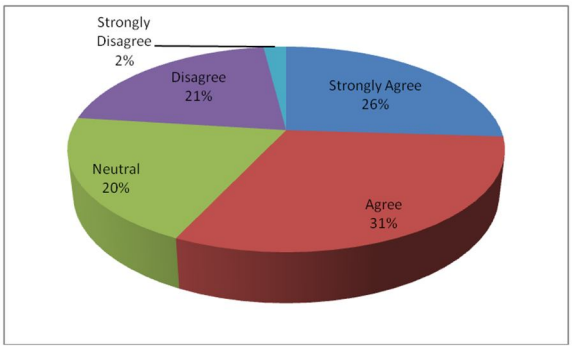
6) The system made it easier to extract key requirements from large specification documents.



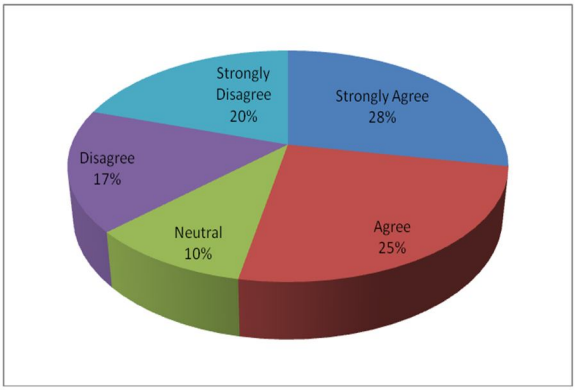
7) The system improved traceability between requirements and source documents.



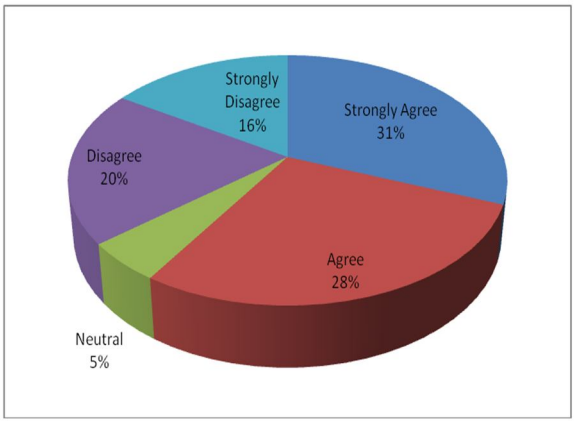
8) Using the system reduced my overall workload during requirements engineering tasks.



9) The privacy and IP protection provided by the local deployment influenced my trust in the system.



10) I would recommend the use of a local LLM + RAG system for requirements engineering in automotive projects.



IX. FINDINGS

This section highlights the results from assessing the local Large Language Model (LLM) combined with a Retrieval-Augmented Generation (RAG) system in the context of automotive requirements engineering. The results are shown from participants' answers to a 10-item Likert scale survey and their actual performance during task completion.

1) Overall Perception of the System: Most of the participants shown the positive perception of the LLM combined with the RAG system. A large number of respondents agreed that the system helped them to work better with complex automotive specification documents. High level of strong agreement on statements about high-speed performance, simple & easier understanding, and with less workload, showing that people saw the system as a very useful for practical requirements of engineering tasks.

2) Perceived Efficiency and Time Savings: Participants consistently provided feedback on the system that significantly helped them to analyze requirements documents more quickly and easily. This was also shown through high agreement scores on statements about faster analysis and simpler extraction of important requirements. When compared these tasks with manually baseline, participants felt that using the LLM-assisted process saved time on finding information, understanding specifications, and creating documents.

3) Accuracy and Reliability of Outputs: The results show that most people thought the system's answers were accurate and dependable. Their responses suggested they had moderate to high confidence in the system's results, especially when the system included sources through the RAG method. though a few participants had mixed opinions, showing some hesitation in fully trusting the system's automated responses.

4) Ambiguity Reduction and Requirement Clarity: A major result from the study was how well the system reduced ambiguity. Participants said the system helped make unclear, incomplete, or messy requirements clearer. Many agreed strongly that the system improved understanding, showing that the LLM combined with RAG worked well in helping interpret and refine requirements.

5) Trust, Privacy, and IP Protection: Participants expressed strong confidence in the system being deployed locally. The results indicate that having the system on-premises increased trust, especially when it came to managing sensitive automotive intellectual property and confidential project information. Most people agreed that ensuring data privacy and protecting intellectual property were key reasons they accepted the system.

6) Traceability and Documentation Support: Responses showed that the system significant improvement on the ability to track requirements back to their original documents. People liked being able to find specific parts of text and references, which helped with checking and confirming things. This feature was especially seen as useful in automotive projects where safety is a top priority.

7) User Acceptance and Future Adoption: In summary, the results show that users strongly approve of the LLM combined with RAG system. Most people said they would suggest the system for use in automotive requirements engineering projects. The findings point to a good chance of being used in real situations, especially in companies that value data security, managing knowledge, and having effective documentation processes.

A. Summary of Key Findings

In summary, the study found that:

1) The LLM + RAG system improved perceived speed and efficiency of requirements analysis.
2) Participants reported higher clarity and reduced ambiguity in requirements.
3) Trust in the system was strengthened by local deployment and privacy preservation.
4) Traceability and documentation quality were seen as significant improvements.
5) The system showed strong potential for adoption in automotive requirements engineering environments.

X. CONCLUSION

This study shows the use of a locally placed Large Language Model (LLM) combined with a Retrieval-Augmented Generation (RAG) system to help with requirements engineering in the automotive sector. The results shows that this method can significantly boost the efficiency and quality of handling automotive specification documents, while keeping data private and protecting intellectual property by running everything on-site.

Following is the comparison with Manual system Vs using LLM+RAG

Table 1 Comparison Manual Vs LLM+RAG system

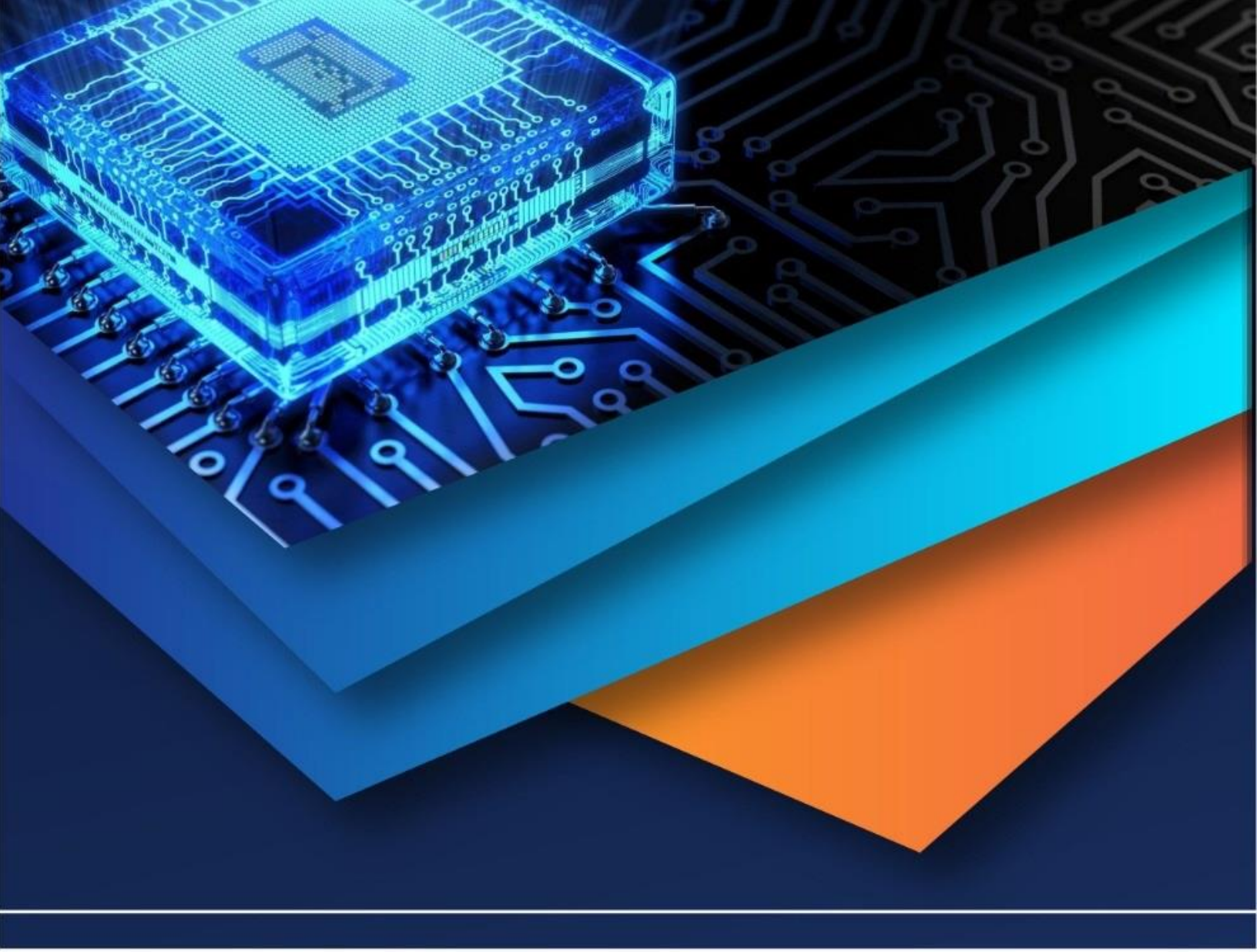
Matric	Manual System	LLM+ RAG	Improvement
Time to extract requirements	42 min	11 min	~74% faster
Accuracy	0.71	0.88	+ 23.9%
Ambiguity detection recall	0.53	0.81	+28%

The findings indicate that using LLM-based processes cut down the time needed to understand requirements and made it easier to interpret complicated or unclear specifications.

People involved in the study felt more confident in the system's accuracy and reliability, especially because the RAG setup allowed them to track information back to its original sources. Their trust in the system was also reinforced by the fact that it was run locally, which helped address important concerns about keeping information secure and following regulations.

REFERENCES

[1] Hull, E., Jackson, K., & Dick, J. (2011). Requirements Engineering. Springer.
[2] Broy, M. (2006). Challenges in automotive software engineering. ICSE.
[3] Lewis, P., et al. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. NeurIPS.
[4] Bommasani, R., et al. (2021). On the opportunities and risks of foundation models. arXiv.
[5] Cleland-Huang, J., et al. (2023). AI-enabled requirements engineering. IEEE Software.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY