frontiers | Frontiers in Computer Science

TYPE Systematic Review
PUBLISHED 27 May 2025
DOI 10.3389/fcomp.2025.1523699

OPEN ACCESS

EDITED BY
Barkaoui Kamel,
Conservatoire National des Arts et Métiers (CNAM), France

REVIEWED BY
Adamantios Koumpis,
University Hospital of Cologne, Germany
Sabina Rossi,
Ca' Foscari University of Venice, Italy
Xiaoding Wang,
Fujian Normal University, China

*CORRESPONDENCE
Manar Abu Talib
✉ mtalib@sharjah.ac.ae

RECEIVED 06 November 2024
ACCEPTED 13 May 2025
PUBLISHED 27 May 2025

CITATION
Saleh Y, Abu Talib M, Nasir Q and Dakalbab F (2025) Evaluating large language models: a systematic review of efficiency, applications, and future directions. *Front. Comput. Sci.* 7:1523699. doi: 10.3389/fcomp.2025.1523699

Evaluating large language models: a systematic review of efficiency, applications, and future directions

Yasmeen Saleh[1], Manar Abu Talib[1]*, Qassim Nasir[2] and Fatima Dakalbab[1]

[1]Department of Computer Science, College of Computing and Informatics, University of Sharjah, Sharjah, United Arab Emirates, [2]Department of Computer Engineering, College of Computing and Informatics, University of Sharjah, Sharjah, United Arab Emirates

Large language models, the innovative breakthrough taking the world by storm, have been applied in several fields, such as medicine, education, finance, and law. Moreover, large language models can integrate into those fields through their abilities in natural language processing, text generation, question answering, and several other use cases that benefit human interactions and decision-making. Furthermore, it is imperative to acknowledge the differences involved with large language models beyond their applications by considering aspects such as their types, setups, parameters, and performance. This could help us understand how each large language model could be utilized to its fullest extent for maximum benefit. In this systematic literature review, we explore each of these aspects in depth. Finally, we conclude with insights and future directions for advancing the efficiency and applicability of large language models.

KEYWORDS

large language models, LLMS, efficiency, performance, application

1 Introduction

In today's world, human interaction with artificial intelligence has significantly risen thanks to the recent advancements in large language models and natural language processing. The field of large language models, while still an emerging subfield of artificial intelligence, is a vast field with varying types and specifications of each large language model and the limitations and accuracies of each. To discover this vast field more, we must develop a basic understanding of large language models, their history, applications, and challenges. Furthermore, efficiency in large language models involves several aspects, including hardware and software requirements, sourcing, training, and output accuracy. Understanding and optimizing the efficiencies of these models is imperative, given the increasing reliance on such technology in various applications. To the best of our knowledge, there are very few Systematic Literature Reviews (SLR) on the efficiencies of large language models, which has motivated this work. Therefore, this systematic literature review aims to provide a comprehensive overview of the state-of-the-art research on the efficiencies of large language models.

Firstly, language models possess the skill of assigning probabilities to sequences of tokens by analyzing statistical patterns in the distribution of a sequence of tokens within data. Modern language models include multiple neural network layers representing tokens within a multidimensional feature space. Unlike early n-gram models that only learned transition probabilities between one-word sequences and the following, neural language models could utilize pre-trained representation of words, called embedding (Bender et al., 2021;

Trott et al., 2023). Since language models cannot store or recall information, having a memory component, such as vector stores, is imperative. Vector stores help search and store embedded data. When data retrieval is required from the vector store, invoked by a user query, the documents could be passed to the large language models (LLMs) through multiple methods. One of the most used methods is called the stuff method. This method is most efficient when passing similar documents in a single prompt, whereas other methods can be used in processing documents that cannot be passed in a single prompt (Topsakal and Akinci, 2023). Sophisticated neural language models with billions of parameters and several deep learning techniques are what modern LLMs are.

Moreover, it is no wonder that with such powerful internal workflow complexity and ease of user access and querying, LLMs will be able to solve a wide range of tasks with complexities while being user-friendly. This sparked the widespread usage and integration of LLMs across various areas and into multiple fields, such as medicine, education, finance, and law. LLMs are essential in disease prediction, diagnosis, and assessment of therapeutic targets in medicine. These include providing treatment guidelines for cancer patients based on their magnetic resonance imaging radionics and predicting aging-related diseases (Singhal et al., 2023; Cascella et al., 2023; Jo et al., 2023). ChatGPT, an LLM, was used to write preauthorization requests for dental insurance companies. Tailored and fine-tuned applications based on LLMs can enhance dental telemedicine services when combined with dental health care personnel (Huang et al., 2023; Eggmann et al., 2023). In education, ChatGPT was used to evaluate student-generated answers in a learning environment and help students generate answers to their questions (Porsdam Mann et al., 2023; Meyer et al., 2023; Kasneci et al., 2023; Milano et al., 2023; Lund et al., 2023). Financial applications include fraud detection, algorithmic trading, and risk assessment (Fan, 2024). In legal settings, LLMs support document analysis, contract review, and automated legal reasoning (Siino et al., 2025). In addition to their established uses in medicine, education, finance, and law, LLMs are being explored in emerging fields such as blockchain. Recent research has highlighted the use of LLMs to automate smart contract verification and improve security in decentralized systems (Ressi et al., 2024). AI-enhanced blockchain technology provides new prospects for boosting trust and accuracy in contract execution, which is still an area for future research.

Unfortunately, despite their various applications, there are still many challenges relating to performance, ethics, and many more. On the ethical front, concerns grow regarding bias and integrity, as these models, developed on extensive data collections, may unknowingly perpetuate and reinforce existing biases present in the training data. This raises questions about the accuracy and fairness of the outputs generated by these models, especially in sensitive applications such as hiring processes or automated decision-making (Head et al., 2023). The extensive knowledge these models acquire while training raises serious privacy problems, raising the possibility of inadvertently disclosing sensitive information and necessitating the implementation of strong privacy protections. Furthermore, the necessity of developing ethical standards to stop malicious use is emphasized by the possibility of manipulating and abusing massive language models to produce false information or participate in disinformation campaigns (Wu et al., 2023).

Beyond ethical concerns, LLMs encounter other challenges. The computational resources required for training and fine-tuning are extensive, limiting access to these technologies for smaller organizations and researchers with constrained computing capabilities. The dependency on training data introduces challenges related to the diversity and quality of the data, potentially leading to difficulties in understanding specific contexts or generating appropriate responses for underrepresented topics. Adapting these models to domain-specific contexts requires careful consideration, as fine-tuning for specialized tasks may be resource-intensive and may only sometimes yield optimal results (Deng et al., 2023). The delicate balance between human-machine collaboration presents a challenge, as it is crucial to ensure that these models augment human capabilities without replacing critical decision-making processes (Bender et al., 2021; Trott et al., 2023).

Continuous learning and updating pose challenges as well. LLMs need frequent updates to stay relevant and accurate, necessitating a robust infrastructure for managing model evolution and ensuring seamless integration with emerging information sources. These challenges underscore the importance of a collaborative effort involving researchers, policymakers, and industry stakeholders to establish ethical guidelines, develop governance mechanisms, and foster responsible use of LLMs. As these models play a transformative role in diverse domains, addressing these challenges is imperative for ensuring ethical and effective social integration (Trott et al., 2023).

With this systematic literature review, we look forward to providing a comprehensive analysis and comparison of the efficiencies of different LLMs. We will contribute to presenting such a comparison by presenting the information we collected on the hardware and software requirements, sourcing, training, and output accuracy associated with these models. This represents a critical step in understanding the multifaceted dimensions of LLM efficiencies, enabling researchers, practitioners, and policymakers to make informed decisions about their utilization and development. By shedding light on the current state of knowledge in this domain, we aim to facilitate the development of accurate and optimal solutions in the era of LLMs.

The remainder of this paper is divided into six sections: Section 2 provides information on related work. Section 3 describes the methodology. Section 4 lists the results and discussions. Section 5 addresses the limitations of this review—finally, Section 6 concludes and suggests suggestions for future work.

2 Related work

During our research, we found a total of 7 survey papers that are related to our topic. These papers have been published in the last 5 years, and most of the documents tackled the advantages, disadvantages, and ethical and legal issues associated with LLMs. Despite our paper discussing similar points, we have mainly focused on the efficiency aspect of LLMs, unlike the other papers. Furthermore, we developed a deeper understanding of LLMs, their efficiencies, application, and overall benefits to compare our work with others. All the papers mentioned below discuss LLMs.

Floridi (2023), Mökander et al. (2023), and Teubner et al. (2023) discuss topics of ethical and legal matters regarding LLMs. To be specific, Floridi (2023) talks about intelligence regarding LLMs. The author provides information regarding LLMs' possible implications and ethical, legal, and human costs. Floridi compares the spiritual, animal, and AI agents and how we interact with them (Floridi, 2023). Secondly, Mökander et al. (2023) delved deeper into implications and discussed auditing, its importance, methods, and limitations. As the

author explained, auditing is the governing process used to recognize and alleviate issues with AI (artificial intelligence) technologies. Auditing LLMs can be done through a three-layered approach, which includes (governance, model, and application) (Mökander et al., 2023). Thirdly, Teubner et al. (2023) discussed the expectations and future involved with LLMs and their implications. Teubner defends LLMs by pointing out that acknowledging their power instead of banning them is a more reasonable action toward LLMs' growth. He also discusses their effectiveness, legality, and threats, clarifying misconceptions and supporting integrating and adopting LLMs into society (Teubner et al., 2023). Fourthly, PLMs (pre-trained language models) and NLP (natural language processing), two fields relating to LLMs, were explored by Min et al. (2023). The survey provides background information on PLMs and categorizes the utilization of PLMs for NLPs into three paradigms: pre-train then fine-tune, prompt-based learning, and NLP as text generation, each discussed in depth (Min et al., 2023). Next, Liu et al. (2023) discuss prompting and provide in-depth background information. The author also explains more complex ideas, such as multi-prompt learning methods and prompt engineering, and provides information on the topic's applications and challenges (Liu et al., 2023). Furthermore, Kamnis (2023) explores GPTs (generative pre-trained transformers) through surface engineering. However, the author's main idea is custom data indexing, which enables entities to organize and store data using AI tools for efficient data retrieval. The author compares GPT-4 and a fine-tuned data-indexed GPT-3 model, evaluating them on their query-answering performances (Kamnis, 2023). Finally, Qureshi et al. (2023) investigate LLMs', specifically ChatGPT's, ability to integrate into SRs (systematic reviews). The author tests ChatGPT's utility and applicability by quizzing it on language interpretation tasks related to systematic reviews. Although ChatGPT faced some challenges, it could still form responses according to what was requested (Qureshi et al., 2023). Floridi (2023) and Teubner et al. (2023) all discuss similar topics of ethical and legal matters regarding LLMs. However, they seem to lack information on the efficiencies of LLMs. Except for Floridi (2023), the other two papers, Mökander et al. (2023) and Teubner et al. (2023), did not include comparisons between LLMs. Similarly, Qureshi et al. (2023) do not conduct comparisons, but they test and discuss topics related to ChatGPT. On the contrary, Kamnis (2023) compares, but the topic is too specific.

In our work, we will conduct a systematic literature review comparing different LLMs focusing on efficiency. Table 1 shows the contributions of each paper. We have added a column describing the difference between our contribution and the others'.

3 Methodology

In this critical review, we used the framework proposed by Kitchenham and Charters methodology to implement our review.

TABLE 1 Summary of related work.

Ref.	Year	Contributions	Difference
Floridi (2023)	2023	Talks about different LLMs, their pros and cons, and their ethical and legal issues.	Provides no information on the requirements or efficiency of the large language models.
Mökander et al. (2023)	2023	This paper analyses and evaluates LLMs from technical, ethical, and legal perspectives. It talks about the opportunities and risks of LLMs, highlights the properties that undermine the feasibility and effectiveness of existing AI auditing procedures, and derives and defends seven claims about how LLM auditing procedures should be designed and how to structure such procedures.	It does not compare large language models or cover the main idea, efficiency.
Teubner et al. (2023)	2023	This paper discusses the emergence of ChatGPT and LLMs in general and their limits, threats, and legality.	It does not directly compare the efficiencies of different large language models and mainly discusses ChatGPT.
Min et al. (2023)	2023	This paper surveys the three trending paradigms that use pre-trained language models for natural language processing. The paper describes each of them in-depth, summarizes prior works whose applications have shown promise, and discusses limitations.	It compares large language models from a natural language processing perspective, but not generally.
Liu et al. (2023)	2023	This paper summarizes and analyses several paradigms in developing statistical natural language processing techniques. It also highlights the commonalities and differences between the four paradigms of natural language processing.	Compares large language models from a prompting parameters perspective.
Kamnis (2023)	2023	This paper demonstrates that a fine-tuned data-indexed GPT model can significantly improve query response performance compared to state-of-the-art GPT-4. This model can provide more accurate, coherent, and relevant responses, which have important implications for developing and applying natural language processing models in surface engineering domains by utilizing domain adaptation and data indexing techniques.	Focuses on large language models, specifically GPT, for surface engineering.
Qureshi et al. (2023)	2023	This paper discusses the capability of ChatGPT and other LLMs and their limitations or reliability in being integrated into systematic reviews.	It only tests ChatGPT and does not test or compare it with other models.

This approach comprises planning, conducting, and reporting phases, each comprising various stages. During the planning phase, a review protocol was formulated, encompassing six stages: articulating research questions, devising the search strategy, delineating study selection procedures, specifying quality assessment rules, outlining the data extraction strategy, and combining the extracted data. Figure 1 illustrates the six stages mentioned.

Figure 1 illustrates our journey from identifying research questions to synthesizing extracted data. The stages involve identifying search terms, searching, initial results, filtering, acquiring the final papers, applying data extraction strategies, finalizing the extraction, and finally synthesizing the extracted data.

Figure 2, shown above, illustrates the process we followed in helping us narrow down our research papers. It first starts with identifying our research questions and search terms. Secondly, we apply an initial search and filtration process. Lastly, we finalize the extraction and double-check if the research technique requires to be repeated.

3.1 Research questions

The formulation of research questions was as follows:

- RQ1: What is the large language model's application and use case deployed?

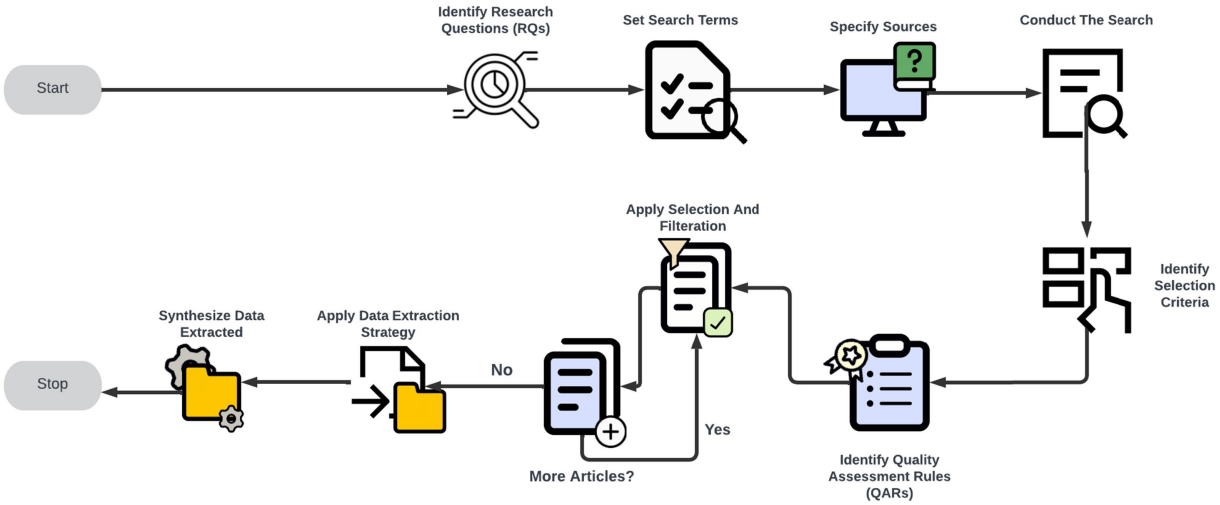

FIGURE 1
The stages of conducting a systematic literature review.

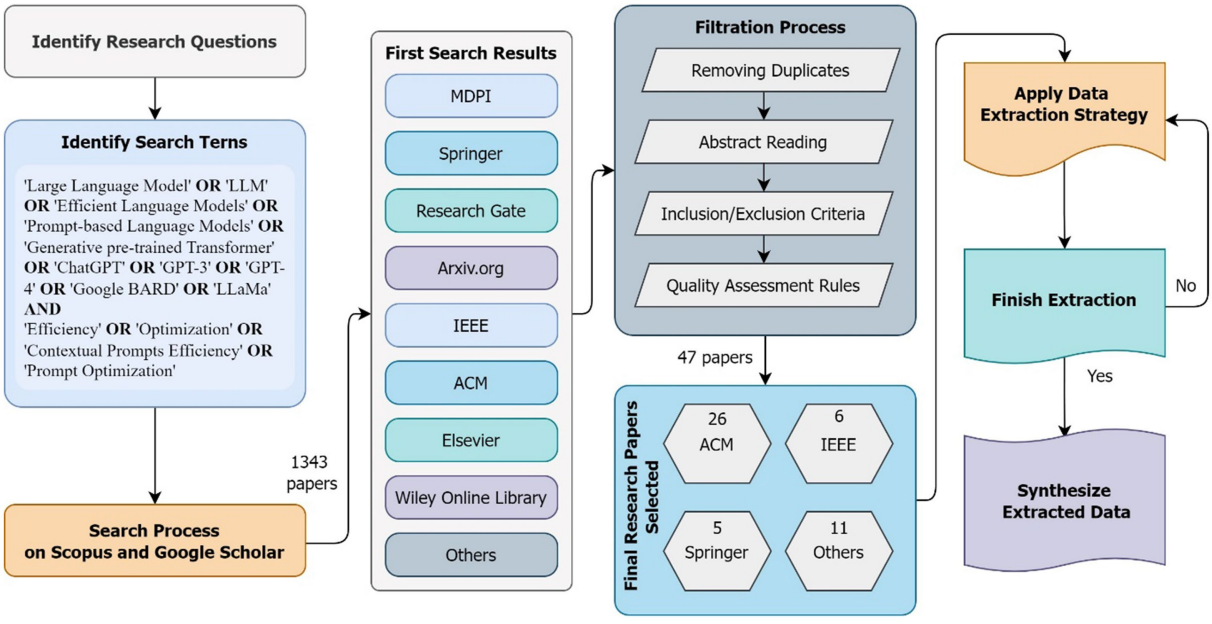

FIGURE 2
Applied research methodology.

This question aims to understand the diverse range of applications where large language models are utilized, shedding light on the practical contexts in which they are deployed.

- RQ2: Which specific type of large language model is employed? Is the considered model open source?

  This question seeks to identify the specific models used in different studies and assess whether they are open-source or proprietary, which can affect replicability and accessibility.

- RQ3: What prerequisites and resource demands are utilized in deploying a large language model? Which hardware specifications were used in the experiment? What were the model parameters employed in the experiment?

  The sub-questions delve into the hardware and computational requirements and the model parameters, providing insights into the resource demands of deploying large language models.

- RQ4: What are the methodologies for assessing the performance metrics of the large language model deployed?

  This question aims to understand the evaluation methods and metrics employed to assess the performance of large language models in various applications, offering insights into their effectiveness and limitations.

3.2 Search strategy

Moving on to the subsequent stage, we provide the search strategy, aligning it with the initial stage to retrieve pertinent articles. Identifying search terms and the leading publishers used, essential for precision in the search, was also addressed.

3.2.1 Key search terms

Table 2, shown below, presents the key search terms used in the search process. These search expressions were identified based on three criteria: Firstly, the research questions were the main driver to guide the determination of the search phrases. Secondly, Boolean operators such as ANDs and ORs were utilized to aid in filtering the search results. Thirdly, new search terms were discovered by exploring relevant resources.

3.2.2 Publishers

1. ACM Digital Library
2. Springer
3. IEEE Explore
4. Elsevier Science Direct
5. Google Scholar

3.3 Study selection

Stage three focused on selection criteria and establishing inclusion and exclusion rules, as shown in Table 3.

3.4 Quality assessment rules (QARs)

In stage 4, we evaluate the collected research articles based on the following QAR set. The QAR utilized in this research are listed below:

QAR 1: Is the application and use case of the deployed large language models stated?
QAR 2: Are the types of large language models used identified and explained?
QAR 3: Are the requirements for deploying the large language model detailed?
QAR 4: Is there a comparison between the efficiency of different large language models?
QAR 5: Is the evaluation of the significant language model/s well performed?
QAR 6: Is the method used for evaluating the large language model clear and accurate?
QAR 7: Are the performance metrics of large language models clearly defined and used?
QAR 8: Is the large language model's experimental setup stated and clear?
QAR 9: Are the large language models' parameters described clearly and concisely?
QAR 10: Does this study provide enough information and evidence to be considered as related to our work?

Each QAR score is allocated based on the following scale. 'Not answered' is assigned a score of 0, 'below average' is valued at 0.25, 'average' is given a score of 0.5, 'above average' is designated 0.75, and 'fully answered' is assigned a score of 1. Each study's overall QAR score was determined on a scale of 1 to 10. Studies that had a score of less than four were disqualified from further synthesis in accordance with our review process. The assessment adhered to a uniform and repeatable methodology that was established during the systematic review's preparation phase, even though the authoring team handled the scoring. This strategy aligns with Kitchenham and Charters' suggestions, which highlight protocol-driven quality evaluation as a means of minimizing bias and improving transparency in software engineering reviews.

3.5 Data extraction strategy

We created a sheet for articles that we found and collected. The sheet includes information regarding the large language model, paper number, paper URL, paper title, author/s, publisher, publisher source, publication type, year of publication, paper description, RQ1(field), RQ2(LLM type, source), RQ3(software requirements, hardware requirements, model parameters), and RQ4(performance metrics). It's imperative to note that not all research papers can answer the research questions.

3.6 Synthesis of extracted data

As emphasized by Kitchenham and Charters, the review protocol holds significant importance in any SLR. Consequently, the authors have held regular meetings to mitigate researcher bias and uphold the quality of the review protocol. Due to the nature of our findings, our

data synthesis technique is qualitative because our RQs do not involve numbers or calculations. In the results and discussions section below, we will organize the data in diagrams to the best of our ability.

4 Results and discussions

This section will discuss the answers to the RQs and their subsections, enabling us to conclude our results for this SLR.

4.1 RQ1: LLM application and use cases

In this research question, we aim to understand what field or area the large language model utilized. Since each paper discussed a different topic in different fields, we created categories to help organize the collected research papers. After studying the papers carefully, we found that most papers covered four fields: data generation (image, text, code, etc.), prompting, modification or control of data (editing, deletion, retrieval, etc.), and prediction. We then illustrated the result of this categorization in Figure 3.

While prompt engineering and data generation may overlap in practice, they are fundamentally independent categories. Designing and organizing input prompts to elicit particular actions or enhance the quality of model output is the primary purpose of prompt engineering. This covers prompt adjustments, prompt templates, and zero−/few-shot instances. Data generation, on the other hand, deals with the output process itself, when new artifacts like text passages, code snippets, or summaries are generated from the LLM. Studies were categorized according to their main goal as stated in each paper: tasks that focused on generating outputs were labeled as data generation, whereas operations that focused on modifying inputs were categorized as prompt engineering. Research papers on LLMs in data generation and prompting appeared most frequently. The references for the documents in each category are listed in Table 4. Out of 27 papers, 13 were related to data generation, nine were prompting-related, four were related to the modification or control of data, and one was associated with data prediction.

TABLE 2 Display of key search terms.

LLM keywords	Operator	Performance keywords
"Large language model" OR "LLM" OR "Efficient language models" OR "Prompt-based language models" OR "Generative pre-trained transformer "OR "ChatGPT" OR "GPT-3" OR "GPT-4" OR "google BARD" OR "LLaMA"	AND	"Efficiency" OR "Optimization" OR "Contextual prompts efficiency" OR "Prompt optimization transformer models."

TABLE 3 Exclusion and inclusion criteria.

Inclusion rules	Exclusion rules
• Trusted source. • Published in the last 5 years. • Direct mention of large language models.	• Weak or unknown source. • Archive (unpublished). • Papers that talk about using large language models, too, specifically. • Papers that do not mention large language models. • Papers less than four on the QAR total score.

4.2 RQ2: LLM type and access

In this research question, we plan to investigate the type of large language model used in the research paper. We also explore whether the LLM is open source or closed source. Figure 4 displays the LLM type along with the frequency; Figure 5 shows whether the LLM type is open or closed source, while Table 5 provides a combination of references for papers involved with each LLM type and source.

From the results in Figure 4 above, we infer that GPT-3 was the most utilized LLM in the research papers studied, with a frequency of 10, meaning it has been used or mentioned by 10 papers. Next comes Codex, with a frequency of 8, making it the second most used or mentioned LLM amongst the papers investigated. Lambda, GPT-3.5, and GPT-2 are all tied with a frequency of 3. Papers (Deng et al., 2023;

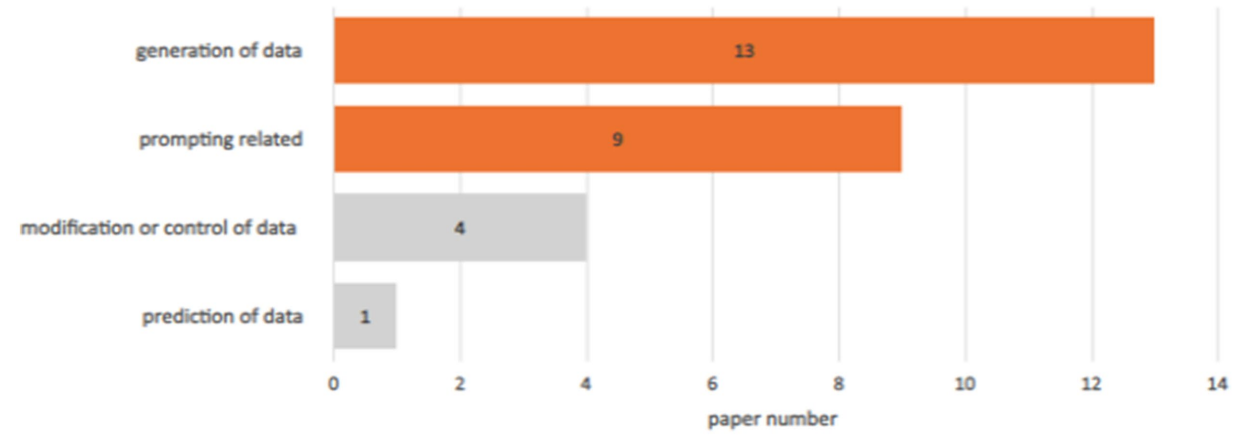

FIGURE 3
Frequency histogram of the 4 LLM categories.

Sarsa et al., 2022; Hämäläinen et al., 2023; Ross et al., 2023; Badini et al., 2023; Mahuli et al., 2023; Macneil et al., 2022; Xu et al., 2022; Strobelt et al., 2023; Wang et al., 2023; Chang, 2023; Zamfirescu-Pereira et al., 2023; Wu et al., 2022; Pan and Ke, 2023; Scells et al., 2023) were papers that either decided to deploy their models or have used models that no other paper used, making the LLM, when represented in a table or illustrated, have a frequency of 1.

Figure 5 answers the question of whether the LLM is open source or closed source. The results display the percentage of papers that utilized open-source LLMs contrasted with those that accessed closed-source LLMs instead. With 59% against 41%, we conclude that most papers used open-source LLMs. This means that out of the 34 LLMs studied carefully in each paper, 20 LLMs were open source, while 14 were closed source. Despite the fact that a number of articles stated the utilization of open-source models, certain fine-tuning techniques were frequently overlooked out or only briefly mentioned. Therefore, it is still difficult to replicate the experimental conditions outlined in those studies. This draws attention to a more general problem in the literature and emphasizes the importance of identifying methods for supporting reproducibility in future research.

Table 5 combines the two previous figures, Figure 4 and 5, and details which papers utilized what specific LLM and whether it was open source.

TABLE 4 Paper reference numbers in each RQ1 category.

Reference	RQ1 Category
Deng et al. (2023); Sarsa et al. (2022); Hämäläinen et al. (2023); Ross et al. (2023); Badini et al. (2023); Mahuli et al. (2023); Macneil et al. (2022); Xu et al. (2022); Jain et al. (2022); Vaithilingam et al. (2022); Di Fede et al. (2022); King (2023); Kang et al. (2023)	Generation of Data
Strobelt et al. (2023); Wang et al. (2023); Chang (2023); Zamfirescu-Pereira et al. (2023); Wu et al. (2022); Jiang et al. (2022); Reynolds and McDonell (2021); Singh et al. (2023); Beurer-Kellner et al. (2023)	Prompting Related
Pan and Ke (2023); Scells et al. (2023); Fan et al. (2023); Urban et al. (2023)	Modification or Control of Data
Kim et al. (2021)	Prediction of Data

4.3 RQ3: Setup approach and LLM parameters

Through this research question, we seek to provide information on the hardware setup utilized for operating the LLM as well as the LLM's parameters. For this question, we considered the variety of LLMs deployed by each research paper and the different resources used by various researchers. Therefore, we will organize the hardware information we collected into smaller components. The results are reflected in Table 6.

Table 6 provides information on the specific hardware components mentioned in the research papers. Although the hardware configurations utilized in the examined research are shown, many of the articles did not provide full system specifications. A variety of configurations are observed among the remaining studies, ranging from high-end multi-GPU systems to more affordable single-GPU or CPU-only setups. This diversity reflects the varying resource capacities of researchers and use cases, and it underscores the need for more consistent reporting in future studies to support improved documentation and analysis of model performance. The most powerful setup among the reviewed papers was reported by the researchers in (Xu et al., 2022). The rest have mixed to lower-end setups, yet they could still deploy powerful LLMs despite that. Most papers still need to provide information regarding the hardware setup.

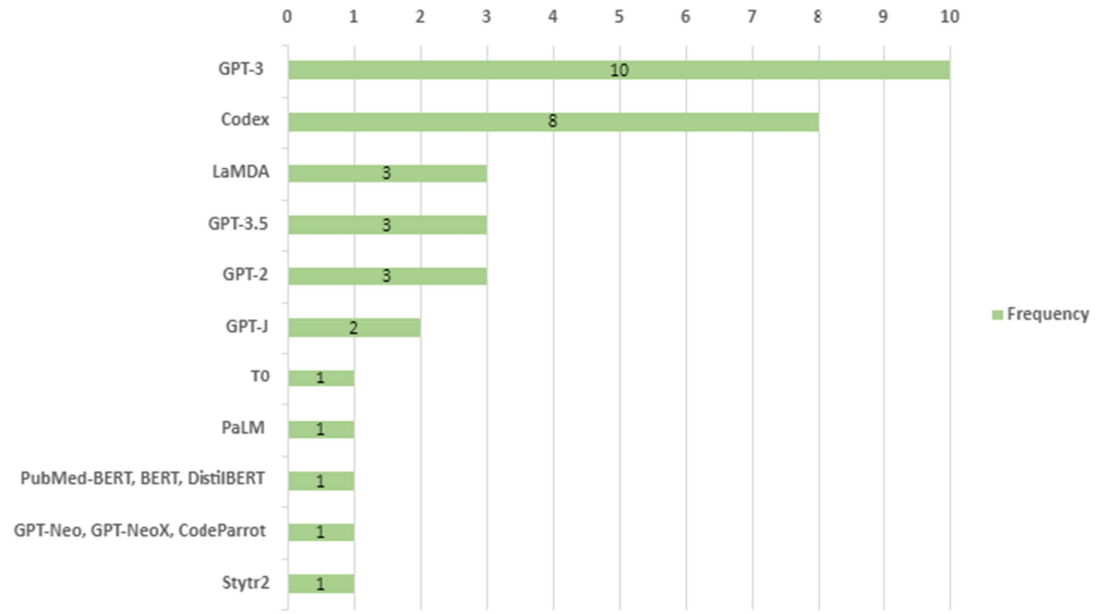

FIGURE 4
Frequency histogram of the LLM types.

For the section about LLM parameter sizes, a figure was generated to show the scale of the models described in the examined research, from largest to smallest. The findings are shown in Figure 6. CuBERT has the fewest parameters (about 345 million), while PaLM has the largest, with 540 billion parameters.

A comprehensive evaluation of deployment feasibility cannot be obtained from parameter count independently; however, it does give a broad idea of model complexity and possible resource requirements. Due to a lack of consistency in the studied literature, important factors, including energy usage during training and inference, as well as the economic cost per inference step, were excluded from our study. In order to provide more comprehensive and useful assessments of model efficiency, this limitation highlights the significance of including energy and cost-related indicators in further studies.

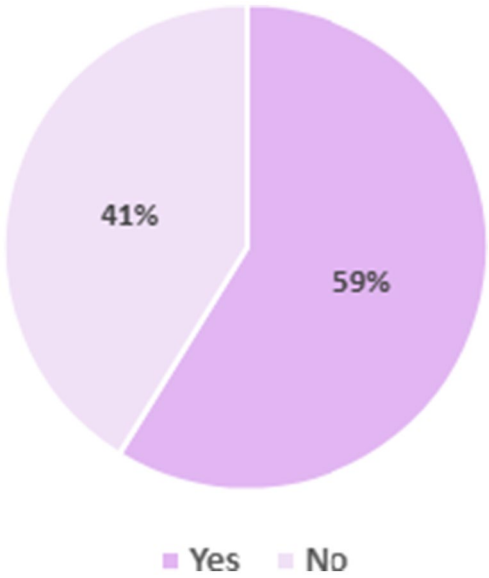

FIGURE 5
Open source LLM frequency.

4.4 RQ4: Performance metrics and evaluation

With this research question, we aim to identify the metrics used to evaluate the performance of different LLMs. It is important to note that the metrics vary from one LLM to another because of the use case or application of the LLM. For example, PaLM (Wang et al., 2023) was evaluated on grammar correctness because the paper is prompting-related. In contrast, Codex (Deng et al., 2023) was considered regarding the number of detected bugs because the paper is related to code generation purpose.

To provide a more structured overview, performance metrics were grouped into six categories: Translation Evaluation Metrics, Code Analysis Metrics, NLP Output Quality Metrics, User Interaction and Feedback Metrics, Model Evaluation Benchmarks, and Domain-Specific Evaluation Metrics. This restructuring addresses differences in evaluation criteria across domains and ensures a more balanced representation, especially for fields like healthcare and education. While code-related metrics remain prominent due to the number of studies in programming contexts, domain-specific metrics have been explicitly highlighted to mitigate cross-domain bias and promote greater clarity. Table 7 presents the reorganized metric categories along with representative evaluation aspects.

TABLE 5 LLM type for each paper.

LLM Frequency			
Reference Number	Large Language Model	Frequency	Open-Source?
Sarsa et al. (2022); Hämäläinen et al. (2023); Macneil et al. (2022); Chang (2023); Jain et al. (2022); Di Fede et al. (2022); Reynolds and McDonell (2021); Singh et al. (2023); Beurer-Kellner et al. (2023); Urban et al. (2023)	GPT-3	10	No
Deng et al. (2023); Sarsa et al. (2022); Ross et al. (2023); Xu et al. (2022); Vaithilingam et al. (2022); Kang et al. (2023); Singh et al. (2023); Fan et al. (2023)	Codex	8	Yes
Wu et al. (2022); Pan and Ke (2023); King (2023)	LaMDA	3	Yes
Badini et al. (2023); Mahuli et al. (2023); Zamfirescu-Pereira et al. (2023)	GPT-3.5	3	No
Xu et al. (2022); Beurer-Kellner et al. (2023); Kim et al. (2021)	GPT-2	3	Yes
Xu et al. (2022); Beurer-Kellner et al. (2023)	GPT-J	2	Yes
Strobelt et al. (2023)	T0	1	Yes
Wang et al. (2023)	PaLM	1	No
Pan and Ke (2023)	Stytr2	1	Yes
Scells et al. (2023)	PubMed-BERT, BERT, DistilBERT	1	Yes
Xu et al. (2022)	GPT-Neo, GPT-NeoX, CodeParrot	1	Yes

TABLE 6 The hardware components of each setup.

Ref	LLM	CPU	RAM	GPUs	Notes
Fan et al. (2023)	Codex	Intel Xeon E5-2660	64GB	1 x NVIDIA Titan V	Single GPU, likely development/testing setup
Beurer-Kellner et al. (2023)	GPT-3, GPT-2, GPT-J	Not available	Not available	1 x NVIDIA A100 (40GB/80GB)	Single A100 GPU, likely inference/smaller models
Deng et al. (2023)	Codex	High-end workstation	256GB	4 x NVIDIA RTX A6000	Powerful multi-GPU setup
Pan and Ke (2023)	Stytr2	Not available	Not available	2 x NVIDIA Tesla P100 + 2 x NVIDIA RTX 3090	Mixed older/newer GPUs, research-specific setup
Scells et al. (2023)	PubMed-BERT, BERT, DistilBERT	Not available	Not available	Not available	No hardware information
Xu et al. (2022)	Codex, GPT-2, GPT-J, GPT-Neo, GPT-NeoX, CodeParrot	Not available	Not available	8 x NVIDIA RTX 8000	Most powerful setup, likely large/complex models/training
Kim et al. (2021)	GPT-2	Not available	Not available	4 x NVIDIA Tesla V100	Multi-GPU setup with older GPUs, research/development
Kang et al. (2023)	Codex	Intel Core i7-7700	32GB	Not available	Low-end setup, likely small models/testing

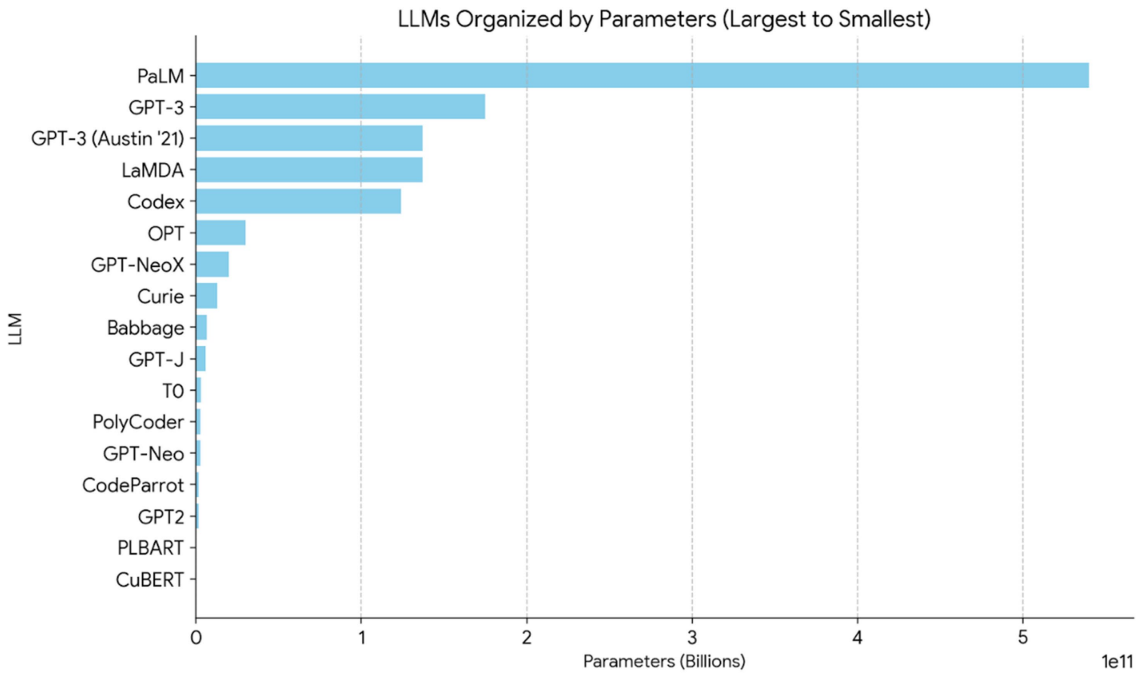

FIGURE 6
Frequency histogram of the LLMs' parameters.

Table 8 reveals the categorization result, presenting each paper with its language model and the information it provided regarding the metrics and category. The table concludes that most papers belong to the "Code Analysis Metrics" category, to be specific, 11 papers evaluated their LLMs on program analysis metrics and natural language processing (NLP) metrics. Next were specific model evaluation metrics, user interaction and feedback metrics, and translation evaluation metrics, with 7, 4, and 1 research papers related to each category in that order.

5 Challenges and recommendations

Over a broad spectrum of applications, LLM has shown considerable promise. Nonetheless, there are several major challenges to overcome, especially in developing fields like smart contract validation, Internet of Things (IoT) integration, and privacy-preserving implementations. To guarantee the safe, open, and responsible application of LLMs in high-stakes situations, these challenges require further consideration.

TABLE 7 Categories of performance metrics and their aspects.

Category	Aspect
Translation Evaluation Metrics	• BLEU (Bilingual Evaluation Understudy): Measures the quality of machine-generated translations by comparing them to reference translations.
Code Analysis Metrics	Program Analysis Metrics: • LOC (Lines of Code): Measures the number of lines of code in a program. • Number of model queries, Number of Decoder calls, Billable tokens: Metrics related to the usage and efficiency of language models in code-related tasks. • Statistical analyses, Frechet Distance, precision, recall, topic similarities and differences, answer consistency, game frequencies: Metrics for evaluating code generation models' statistical properties and performance. • Number of detected bugs, Code coverage, Number of covered APIs, Number of unique, valid programs generated, Execution time: Metrics related to the quality, coverage, and efficiency of generated code.
NLP Output Quality Metrics	• Grammar Correctness, UI Relevance, Question Coverage, BLEU, CIDEr, ROUGE-L, METEOR. • Exact Matches, Contains GT, Sub-String of GT, Micro-F1: Metrics related to the quality and relevance of natural language outputs, especially in conversational contexts.
User Interaction and Feedback Metrics	• Success rate (SR), goal conditions recall (GCR), Executability (Exec): Metrics measuring the success and effectiveness of user interactions with language models. • Quantitative and qualitative participant feedback, Surveys: Metrics involving user feedback, satisfaction, and perception. • Number of errors encountered during task completion, Number of retries required to complete a task, Time taken to complete a task, Perceived ease of use, and usefulness of the tool: Metrics assessing the user experience, efficiency, and usability of language models.
Model Evaluation Benchmarks	• GLUE (General Language Understanding Evaluation Benchmark): Evaluates a model's performance on various NLP tasks. • CRIT (Critical Reading Inquisitive Template): Evaluates models based on critical reading comprehension. • Perplexity: Measures how well a language model predicts the next token in a sequence of code or text. • Recall, precision, f-measure: Standard metrics for evaluating the performance of models in information retrieval or classification tasks.
Domain-Specific Evaluation Metrics	• Using ROBINS-I tool and Risk of Bias analysis, Data extraction from a randomized controlled trial: Metrics related to evaluating research studies and experiments. • Likert Scale: Measures attitudes or opinions using a scale of responses.

The use of LLMs to support smart contract verification is an emerging area of interest. Although LLMs can assist in developing, summarizing, or analyzing smart contracts, their integration within blockchain systems presents unique difficulties. Smart contracts demand precise and verifiable logic, where even minor errors can lead to significant financial consequences. The limited interpretability of LLM-generated outputs further complicates efforts to trace and validate contract code, particularly in security-critical scenarios. Significant privacy and security issues also arise from the integration of LLMs into edge computing and industrial IoT environments. One study, Wang et al. (2022) emphasizes the importance of secure data aggregation techniques in blockchain-enabled IoT systems to protect user privacy. Another work, Wang et al. (2022) highlights the complexity of developing hierarchical trust evaluation models in 5G-enabled intelligent transportation systems, especially when incorporating AI-driven components like LLMs. Additionally, hierarchical federated learning has demonstrated the potential to enhance both privacy and anomaly detection in industrial settings (Wang et al., 2023). Collectively, these studies underscore the urgency of adapting LLM deployments to privacy-aware architectures, particularly when dealing with real-time, sensitive, or decentralized data.

Future LLM applications in smart contracts should include formal verification tools that confirm logic soundness and identify potential vulnerabilities in order to address these problems. To reduce data exposure during model training and inference, techniques like differential privacy and federated learning should be further investigated in privacy-sensitive domains like the IoT and transportation. Furthermore, policy and regulatory frameworks need to change to take into account the increasing role that LLMs play in operational, financial, and legal decision-making. Lastly, to improve reproducibility and ease cross-domain benchmarking, researchers are encouraged to accept stronger reporting standards, especially with regard to fine-tuning methods, evaluation processes, and deployment.

6 Conclusion and future work

In our systematic literature review, we researched a comparison between large language models, with our focus on their efficiency. We reviewed 27 research papers published between 2019 and 2023. We also crafted four research questions that we believed would be relevant in helping with our comparison. RQ1 covered the field the LLM was used in, RQ2 covered the type of LLM as well as whether it is open source or not, RQ3 covered hardware requirements as well as the LLMs' parameters, and finally, RQ4 covered the metrics used for the evaluation of the LLM. We collected research papers and evaluated them based on the above research questions.

Our findings revealed that most studies leveraged LLMs for data generation tasks, followed by prompting-related applications. GPT-3 was the most widely used model, appearing in 10 studies, followed by Codex. A majority of studies utilized open-source LLMs, while others employed proprietary models. Our analysis of hardware setups highlighted a lack of detailed reporting on computational resources, though one study utilized an 8 x NVIDIA RTX 8000 GPU setup for high-performance LLM deployment. Regarding evaluation, we observed a strong emphasis on code analysis metrics, followed by model-specific evaluations and user interaction feedback.

TABLE 8 LLM performance metrics.

Ref	Model	Metrics	Category
Reynolds and McDonell (2021)	GPT-3	BLEU (French-to-English translations)	Translation Evaluation Metrics
Fan et al. (2023)	Codex	Manual analysis, codex-e, TBar, and Recorder	Code Analysis Metrics
Beurer-Kellner et al. (2023)	GPT-3, GPT-2, GPT-J	LOC, Number of model queries, Number of Decoder calls, Billable tokens	Code Analysis Metrics
Sarsa et al. (2022)	GPT-3, Codex	Programmatic analysis	Code Analysis Metrics
Hämäläinen et al. (2023)	GPT-3	Statistical analyses, Frechet Distance, precision, recall, topic similarities and differences, answer consistency, game frequencies	Code Analysis Metrics
Deng et al. (2023)	Codex	Number of detected bugs, Code coverage, Number of covered APIs, Number of unique, valid programs generated, Execution time	Code Analysis Metrics
Wang et al. (2023)	PaLM	Grammar Correctness, UI Relevance, Question Coverage, BLEU, CIDEr, ROUGE-L, and METEOR, Exact Matches, Contains GT, Sub-String of GT, Micro-F1	Code Analysis Metrics
Scells et al. (2023)	PubMed-BERT, BERT, DistilBERT	Recall, precision, f-measure	Code Analysis Metrics
Zamfirescu-Pereira et al. (2023)	GPT-3.5	Number of errors encountered during task completion, Number of retries required to complete a task, Time taken to complete a task, Perceived ease of use, and usefulness of the tool	User Interaction and Feedback Metrics
Badini et al. (2023)	GPT-3.5	Resolution of specific 3D printing issues considering filament material and other conditions	Code Analysis Metrics
Wu et al. (2022)	LaMDA	Likert Scale, Interaction mechanisms and behaviors, Consecutive run, Edited, Curated, Created, Undone	User Interaction and Feedback Metrics
Xu et al. (2022)	Codex, GPT-2, GPT-J, GPT-Neo, GPT-NeoX, CodeParrot	Perplexity, Code completion accuracy, Human evaluation	Code Analysis Metrics
Singh et al. (2023)	Codex	Success rate (SR), goal conditions recall (GCR), Executability (Exec)	User Interaction and Feedback Metrics
Ross et al. (2023)	Codex	Quantitative and qualitative feedback from 42 participants, Surveys (pre-study, pre-task, post-task)	User Interaction and Feedback Metrics
Mahuli et al. (2023)	GPT-3.5	Using the ROBINS-I tool and Risk of Bias analysis, Data extraction from a randomized controlled trial	User Interaction and Feedback Metrics
Strobelt et al. (2023)	T0	GLUE (General Language Understanding Evaluation Benchmark)	Specific Model Evaluation Metrics
Chang (2023)	GPT-3	CRIT (Critical Reading Inquisitive Template)	Specific Model Evaluation Metrics
Macneil et al. (2022)	GPT-3	Tracing the execution of code, Fixing bugs, Explaining how they were fixed, Generating analogies, Listing relevant programming concepts, Predicting the console output	Specific Model Evaluation Metrics
King (2023)	LaMDA	Analysis of the accuracy of scientific references generated by Google's Bard chatbot	Specific Model Evaluation Metrics
Urban et al. (2023)	GPT-3	The accuracy of natural language prompts and structured prompts	Specific Model Evaluation Metrics
Kim et al. (2021)	GPT-2	Evaluation of next token prediction for leaf tokens	Specific Model Evaluation Metrics
Kang et al. (2023)	Codex	"-acc@n," precision, wef, wef@n	Specific Model Evaluation Metrics

Future studies should investigate more detailed efficiency indicators, including delay inference, energy usage, and model resilience, alongside computing cost and accuracy. Comparative studies in fields such as law, finance, and scientific research could provide further insights into the specialized performance of LLMs. Addressing biases, data privacy challenges, and adversarial robustness will require more systematic evaluations. Additionally, advancements in model optimization techniques, such as pruning, quantization, and efficient fine-tuning, can help mitigate the computational burden of large-scale deployment. Beyond efficiency, future research should emphasize interpretability and usability, as these factors are crucial for real-world adoption. Transparency in decision-making, bias reduction, and explainability in high-risk applications, particularly in healthcare and finance, remain critical research areas. Exploring LLM applications in novel domains, such as blockchain-based smart contract verification, could further reveal insights into their adaptability and security implications. By addressing these research gaps, the continued evolution of LLMs can be guided toward maximizing efficiency while mitigating potential risks associated with widespread deployment.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

YS: Data curation, Formal analysis, Methodology, Resources, Visualization, Writing – original draft. MA: Data curation, Funding acquisition, Project administration, Resources, Software, Supervision, Writing – review & editing. QN: Conceptualization, Formal analysis, Investigation, Project administration, Resources, Supervision, Writing – review & editing. FD: Conceptualization, Data curation, Formal analysis, Investigation, Resources, Visualization, Writing – original draft, Writing – review & editing.

Funding

The author(s) declare that no financial support was received for the research and/or publication of this article.

Acknowledgments

We like to convey our sincere appreciation to the General Civil Aviation Authority (GCAA) of the UAE for founding the Aerospace Centre of Excellence and executing this research study. We express our gratitude to our supervisors and colleagues from the OpenUAE Research and Development Group at the University of Sharjah for their invaluable insights and knowledge that significantly contributed to the research.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declare that no Gen AI was used in the creation of this manuscript.

Publisher's note

References

Badini, S., Regondi, S., Frontoni, E., and Pugliese, R. (2023). Assessing the capabilities of ChatGPT to improve additive manufacturing troubleshooting. *Adv. Indust. Eng. Polymer Res.* 6, 278–287. doi: 10.1016/j.aiepr.2023.03.003

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S., (2021). "On the dangers of stochastic parrots: can language models be too big?," in *FAccT 2021 - Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. pp. 610–623.

Beurer-Kellner, L., Fischer, M., and Vechev, M. (2023). Prompting is programming: a query language for large language models. *Proc. ACM Program. Lang.* 7, 1946–1969. doi: 10.1145/3591300

Cascella, M., Montomoli, J., Bellini, V., and Bignami, E. (2023). Evaluating the feasibility of ChatGPT in healthcare: an analysis of multiple clinical and research scenarios. *J. Med. Syst.* 47, 1–5. doi: 10.1007/S10916-023-01925-4/TABLES/2

Chang, E. Y., (2023). "Prompting large language models with the Socratic method," In *2023 IEEE 13th Annual Computing and Communication Workshop and Conference, CCWC 2023*, Institute of Electrical and Electronics Engineers Inc., pp. 351–360.

Deng, Y., Xia, C. S., Peng, H., Yang, C., and Zhang, L., (2023). "Large language models are zero-shot Fuzzers: fuzzing deep-learning libraries via large language models," *ISSTA 2023 - Proceedings of the 32nd ACM SIGSOFT International Symposium on Software Testing and Analysis*. pp. 423–435.

Di Fede, G., Rocchesso, D., Dow, S. P., and Andolina, S., (2022). "The idea machine: LLM-based expansion, rewriting, combination, and suggestion of ideas," in *ACM International Conference Proceeding Series*. pp. 623–627.

Eggmann, F., Weiger, R., Zitzmann, N. U., and Blatz, M. B. (2023). Implications of large language models such as ChatGPT for dental medicine. *J. Esthet. Restor. Dent.* 35, 1098–1102. doi: 10.1111/jerd.13046

Fan, M., (2024). "LLMs in banking: applications, challenges, and approaches," in *Proceedings of the International Conference on Digital Economy, Blockchain and Artificial Intelligence*, in DEBAI '24. New York, NY, USA: Association for Computing Machinery. pp. 314–321.

Fan, Z., Gao, X., Mirchev, M., Roychoudhury, A., and Tan, S. H., (2023). "Automated repair of programs from large language models," Institute of Electrical and Electronics Engineers (IEEE). pp. 1469–1481.

Floridi, L. (2023). AI as agency without intelligence: on ChatGPT, large language models, and other generative models. *Philos. Technol.* 36, 1–7. doi: 10.1007/S13347-023-00621-Y/FIGURES/3

Hämäläinen, P., Tavast, M., and Kunnari, A., (2023). "Evaluating large language models in generating synthetic HCI research data: a case study," in *Conference on Human Factors in Computing Systems - Proceedings*.

Head, C. B., Jasper, P., McConnachie, M., Raftree, L., and Higdon, G. (2023). Large language model applications for evaluation: opportunities and ethical implications. *N. Dir. Eval.* 2023, 33–46. doi: 10.1002/EV.20556

Huang, H., Zheng, O., Wang, D., Yin, J., Wang, Z., Ding, S., et al. (2023). ChatGPT for shaping the future of dentistry: the potential of multi-modal large language model. *Int. J. Oral Sci.* 15, 29–13. doi: 10.1038/s41368-023-00239-y

Jain, N., Vaidyanath, S., Iyer, A., Natarajan, N., Parthasarathy, S., Rajamani, S., et al. (2022). "Jigsaw: large language models meet program synthesis," in *Proceedings - International Conference on Software Engineering, IEEE Computer Society*. pp. 1219–1231.

Jiang, E., Olson, K., Toh, E., Molina, A., Donsbach, A., Terry, M., et al. (2022). "PromptMaker: prompt-based prototyping with large language models," in *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery.

Jo, E., Epstein, D. A., Jung, H., and Kim, Y. H., (2023). "Understanding the benefits and challenges of deploying conversational AI leveraging large language models for public health intervention," *Conference on Human Factors in Computing Systems - Proceedings*. p. 16.

Kamnis, S. (2023). Generative pre-trained transformers (GPT) for surface engineering. *Surf. Coat. Technol.* 466:129680. doi: 10.1016/j.surfcoat.2023.129680

Kang, S., Yoon, J., and Yoo, S., (2023). "Large language models are few-shot testers: exploring LLM-based general bug reproduction," in *Proceedings - International Conference on Software Engineering*. pp. 2312–2323.

Kasneci, E., Sessler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., et al. (2023). ChatGPT for good? On opportunities and challenges of large language models for education. *Learn. Individ. Differ.* 103:102274. doi: 10.1016/j.lindif.2023.102274

Kim, S., Zhao, J., Tian, Y., and Chandra, S., (2021). "Code prediction by feeding trees to transformers," *Proceedings - International Conference on Software Engineering*, pp. 150–162.

King, M. R. (2023). Can Bard, Google's experimental Chatbot based on the LaMDA large language model, help to analyze the gender and racial diversity of authors in your cited scientific references? *Cell. Mol. Bioeng.* 16, 175–179. doi: 10.1007/s12195-023-00761-3

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., and Neubig, G. (2023). Pre-train, prompt, and predict: a systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.* 55, 1–35. doi: 10.1145/3560815

Lund, B. D., Wang, T., Mannuru, N. R., Nie, B., Shimray, S., and Wang, Z. (2023). ChatGPT and a new academic reality: artificial intelligence-written research papers and the ethics of the large language models in scholarly publishing. *J. Assoc. Inf. Sci. Technol.* 74, 570–581. doi: 10.1002/ASI.24750

Macneil, S., Tran, A., Mogil, D., Bernstein, S., Ross, E., and Huang, Z., (2022). "Generating diverse code explanations using the GPT-3 large language model," in *Proceedings of the 2022 ACM Conference on International Computing Education Research - Volume 2*. p. 3.

Mahuli, S. A., Rai, A., Mahuli, A. V., and Kumar, A. (2023). Application ChatGPT in conducting systematic reviews and meta-analyses. *Br. Dent. J.* 235, 90–92. doi: 10.1038/s41415-023-6132-y

Meyer, J. G., Urbanowicz, R. J., Martin, P. C. N., O'Connor, K., Li, R., Peng, P. C., et al. (2023). ChatGPT and large language models in academia: opportunities and challenges. *BioData Mining* 16, 20–11. doi: 10.1186/s13040-023-00339-9

Milano, S., McGrane, J. A., and Leonelli, S. (2023). Large language models challenge the future of higher education. *Nat. Mach. Intellig.* 5, 333–334. doi: 10.1038/s42256-023-00644-2

Min, B., Ross, H., Sulem, E., Veyseh, A. P. B., Nguyen, T. H., Sainz, O., et al. (2023). Recent advances in natural language processing via large pre-trained language models: a survey. *ACM Comput. Surv.* 56, 1–40. doi: 10.1145/3605943

Mökander, J., and Schuett, J., · Hannah, R. Kirk, and Floridi, Luciano, "Auditing large language models: a three-layered approach," *AI Ethics* vol. 1, pp. 1–31, (2023). doi: 10.1007/S43681-023-00289-2

Pan, B., and Ke, Y. K., (2023). "Efficient artistic image style transfer with large language model (LLM): a new perspective," in *Proceedings of the 8th International Conference on Communication and Electronics systems, ICCES 2023*, Institute of Electrical and Electronics Engineers Inc. pp. 1729–1732.

Porsdam Mann, S., Earp, B. D., Møller, N., Vynn, S., and Savulescu, J. (2023). AUTOGEN: a personalized large language model for academic enhancement—ethics and proof of principle. *Am. J. Bioeth.* 23, 28–41. doi: 10.1080/15265161.2023.2233356

Qureshi, R., Shaughnessy, D., Gill, K. A. R., Robinson, K. A., Li, T., and Agai, E. (2023). Are ChatGPT and large language models 'the answer' to bringing us closer to systematic review automation? *Syst. Rev.* 12, 1–4. doi: 10.1186/S13643-023-02243-Z/PEER-REVIEW

Ressi, D., Romanello, R., Piazza, C., and Rossi, S. (2024). AI-enhanced blockchain technology: a review of advancements and opportunities. *J. Netw. Comput. Appl.* 225:103858. doi: 10.1016/j.jnca.2024.103858

Reynolds, L., and McDonell, K., (2021). "Prompt programming for large language models: beyond the few-shot paradigm," in *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery.

Ross, S. I., Martinez, F., Houde, S., Muller, M., and Weisz, J. D., (2023). "The Programmer's assistant: conversational interaction with a large language model for software development," in *International Conference on Intelligent User Interfaces, Proceedings IUI*, Association for Computing Machinery. pp. 491–514.

Sarsa, S., Denny, P., Hellas, A., and Leinonen, J., (2022). "Automatic generation of programming exercises and code explanations using large language models," in *ICER 2022 - Proceedings of the 2022 ACM Conference on International Computing Education Research*. Association for Computing Machinery, Inc. pp. 27–43.

Scells, H., Schlatt, F., and Potthast, M., (2023). "Smooth operators for effective systematic review queries," in *SIGIR 2023 - Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Association for Computing Machinery, Inc pp. 580–590.

Siino, M., Falco, M., Croce, D., and Rosso, P. (2025). Exploring LLMs applications in law: a literature review on current legal NLP approaches. *IEEE Access* 13, 18253–18276. doi: 10.1109/ACCESS.2025.3533217

Singh, I., Blukis, V., Mousavian, A., Goyal, A., Xu, D., Tremblay, J., et al. (2023). "ProgPrompt: generating situated robot task plans using large language models," in *Proceedings - IEEE International Conference on Robotics and Automation*, Institute of Electrical and Electronics Engineers Inc. pp. 11523–11530.

Singhal, K., Azizi, S., Tu, T., Mahdavi, S. S., Wei, J., Chung, H. W., et al. (2023). Large language models encode clinical knowledge. *Nature* 620, 172–180. doi: 10.1038/s41586-023-06291-2

Strobelt, H., Webson, A., Sanh, V., Hoover, B., Beyer, J., Pfister, H., et al. (2023). Interactive and visual prompt engineering for ad-hoc task adaptation with large language models. *IEEE Trans. Vis. Comput. Graph.* 29, 1146–1156. doi: 10.1109/TVCG.2022.3209479

Teubner, T., Flath, C. M., Weinhardt, C., van der Aalst, W., and Hinz, O. (2023). Welcome to the era of ChatGPT et al.: the prospects of large language models. *Bus. Inf. Syst. Eng.* 65, 95–101. doi: 10.1007/s12599-023-00795-x

Topsakal, O., and Akinci, T. C. (2023). Creating large language model applications utilizing LangChain: a primer on developing LLM apps fast. *Int. Conf. Appl. Eng. Nat. Sci.* 1, 1050–1056. doi: 10.59287/icaens.1127

Trott, S., Jones, C., Chang, T., Michaelov, J., and Bergen, B. (2023). Do large language models know what humans know? *Cogn. Sci.* 47:13309. doi: 10.1111/COGS.13309

Urban, M., Nguyen, D. D., and Binnig, C., (2023). "OmniscientDB: a large language model-augmented DBMS that knows what other DBMSs do not know," *Proceedings of the 6th International Workshop on Exploiting Artificial Intelligence Techniques for Data Management, aiDM 2023 - in conjunction with the 2023 ACM SIGMOD/PODS Conference*.

Vaithilingam, P., Zhang, T., and Glassman, E. L., (2022). "Expectation vs. experience: evaluating the usability of code generation tools powered by large language models," in *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery.

Wang, X., Garg, S., Lin, H., Hu, J., Kaddoum, G., Jalil Piran, M., et al. (2022). Toward accurate anomaly detection in industrial internet of things using hierarchical federated learning. *IEEE Internet Things J.* 9, 7110–7119. doi: 10.1109/JIOT.2021.3074382

Wang, X., Garg, S., Lin, H., Kaddoum, G., Hu, J., and Hassan, M. M. (2023). Heterogeneous Blockchain and AI-driven hierarchical trust evaluation for 5G-enabled intelligent transportation systems. *IEEE Trans. Intell. Transp. Syst.* 24, 1–10. doi: 10.1109/TITS.2021.3129417

Wang, X., Garg, S., Lin, H., Kaddoum, G., Hu, J., and Hossain, M. S. (2022). A secure data aggregation strategy in edge computing and Blockchain-empowered internet of things. *IEEE Internet Things J.* 9, 14237–14246. doi: 10.1109/JIOT.2020.3023588

Wang, B., Li, G., and Li, Y., (2023). "Enabling conversational interaction with Mobile UI using large language models," in *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery.

Wu, X., Duan, R., and Ni, J. (2023). Unveiling security, privacy, and ethical concerns of ChatGPT. *J. Inform. Intellig.* 2, 102–115. doi: 10.1016/J.JIIXD.2023.10.007

Wu, T., Terry, M., and Cai, C. J., (2022). "AI chains: transparent and controllable human-AI interaction by chaining large language model prompts," in *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery.

Xu, F. F., Alon, U., Neubig, G., and Hellendoorn, V. J., (2022). "A systematic evaluation of large language models of code," in *Association for Computing Machinery (ACM)*. pp. 1–10.

Zamfirescu-Pereira, J. D., Wong, R. Y., Hartmann, B., and Yang, Q., (2023). "Why Johnny Can't prompt: how non-AI experts try (and fail) to design LLM prompts," in *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery.