

Domain-specific knowledge and context in large language models: challenges, concerns, and solutions

Kiran Mayee Adavala¹, Om Adavala²

¹Department of Computer Science and Engineering (AI and ML), Kakatiya Institute of Technology and Science, Warangal, India

²School of Cyber Security and Digital Forensics, National Forensic Sciences University, Gandhinagar, India

Article Info

Article history:

Received Sep 20, 2024

Revised Mar 30, 2025

Accepted Jun 11, 2025

Keywords:

Bias

Contextual understanding

Domain-specific knowledge

Expert annotations

Large language models

Memory augmented models

Transfer learning

ABSTRACT

Large language models (LLMs) are ubiquitous today with major usage in the fields of industry, research, and academia. LLMs involve unsupervised learning with large natural language data, obtained mostly from the internet. There are several challenges that arise because of these data sources. One such challenge is with respect to domain-specific knowledge and context. This paper deals with the major challenges faced by LLMs due to data sources, such as, lack of domain expertise, understanding specialized terminology, contextual understanding, data bias, and the limitations of transfer learning. This paper also discusses some solutions for the mitigation of these challenges such as pre-training LLMs on domain-specific corpora, expert annotations, improving transformer models with enhanced attention mechanisms, memory-augmented models, context-aware loss functions, balanced datasets, and the use of knowledge distillation techniques.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Kiran Mayee Adavala

Department of Computer Science and Engineering (AI and ML), Kakatiya Institute of Technology and Science
Warangal, Telangana, India

Email: ak.csm@kitsw.ac.in

1. INTRODUCTION

Large language models or LLMs are a recent disruption involving intelligent answering tools that extract knowledge from internet-related sources such as web pages, research papers, and publicly available knowledge-bases such as Wikipedia. LLMs have some major components that enable them to assimilate queries and generate solutions in a simple and understandable manner. The various components of LLMs are presented in Figure 1 in cyclic order of their usage. The input embedding layer performs tokenization and lookup, similar to any machine text-processing. In the second step, positional encodings are used for saving order of tokens. Thirdly, transformer layers assign weights based on importance, apply feed-forward (neural network) model to normalize the outputs, and finally implement residual connections for stabilization. After this, stacked transformer blocks enable the model to build progressively complex text representations. Next, a linear output layer using SoftMax function predicts or generates the next token. This is followed by cross-entropy, which is used for training. Methods such as Adam or, the more recent AdamW are used to adjust model parameters to minimize the loss function in training. Contextual dependencies are captured using self-attention and cross-attention mechanisms.

Humongous corpus is used in pre-training to learn general language patterns. The model is pre-trained on specific domain datasets for specialization. The model is also made aware of positions of each token in a sequence using positional encoding and relative position representations. The problem of over-fitting is prevented by implementing dropout layers, weight decay, or label smoothing in every update step. The result is a powerful LLM, capable of understanding and generating text with high coherence and

contextual relevance. Some of the popular LLMs are generative pre-trained transformer (GPT) from OpenAI, bidirectional encoder representations from transformers (BERT) from Google, text-to-text transfer transformer (T5) from Google, XLNet from Google-CMU, robustly optimized BERT approach (RoBERTa) from Facebook AI, a lite BERT (ALBERT) from Google-TTIC, Turing-natural language generation (Turing-NLG) from Microsoft, enhanced representation through knowledge integration (ERNIE) from Baidu, Megatron language model (Megatron-LM) from NVIDIA, and DeepSeek from high-flyer.

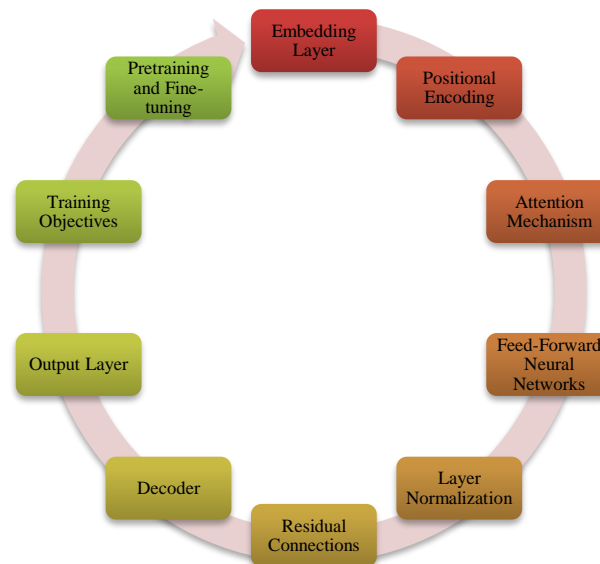


Figure 1. Components of LLM in order of their usage

The current state-of-the-art in LLMs is marked by models like GPT-4, PaLM 2, LLaMA, DeepSeek, and Gemini, which showcase breakthroughs in natural language understanding and generation. These models have become highly proficient in tasks such as text generation, translation, summarization, and complex reasoning. DeepSeek is a specialized model for enhanced search capabilities, offering more context-aware and relevant responses in search queries. Gemini, developed by Google DeepMind, integrates language models with multimodal capabilities, handling both text and visual inputs to deliver highly accurate responses across diverse tasks. Additionally, models are being scaled to trillions of parameters, improving performance with fewer resources. The integration of reinforcement learning from human feedback (RLHF) enhances reliability and ethical safeguards. Research is also focused on improving the models' safety, bias reduction, and interactivity, paving the way for more versatile and responsible AI tools.

2. LITERATURE SURVEY

The introduction of transformer architecture has been foundational for many subsequent LLMs [1]. Among these, BERT is a widely used LLM architecture that significantly advances the state-of-the-art in natural language understanding tasks [2]. Radford *et al.* [3] introduce the GPT architecture, which demonstrates the effectiveness of autoregressive language modeling for generating coherent text. Radford *et al.* [4] also present the more efficient GPT2 model, highlighting its large scale and ability to perform a wide range of language tasks. Yang *et al.* [5] propose XLNet, a novel autoregressive language model that overcomes limitations of BERT by leveraging permutations during pre-training and integrating ideas from transformer-XL. Liu *et al.* [6] introduce RoBERTa, with improved performance over BERT by optimizing training hyper-parameters and using larger datasets. Sun *et al.* [7] present ERNIE 2.0, which extends BERT with a continual pre-training framework to adapt to new tasks and domains. Keskar *et al.* [8] present conditional transformer language (CTRL), a conditional language model designed for controllable text generation by analyzing large volumes of data using model-based source attribution. Raffel *et al.* [9] propose T5, which utilizes transfer learning and frames all natural language processing (NLP) tasks as text-to-text problems, resulting in achieving state-of-the-art results on a wide range of benchmarks. Raiaan *et al.* [10] provide a comprehensive overview of current LLMs. Ge *et al.* [11] introduce OpenAGI for real-world tasks. Huang *et al.* [12] present domain specific question answering language model (DSQA-LLM) for informative

domain-specific queries. Similarly, Sipio *et al.* [13] explore the role of LLMs in the extraction of language semantics. Holtzman *et al.* [14] investigate issues related to degenerate text generation in autoregressive language models and propose strategies to mitigate it. Bodor *et al.* [15] leverages LLMs for data enrichment and monitoring towards performance optimization.

Lewis *et al.* [16] introduce bidirectional and auto-regressive transformers (BART), a sequence-to-sequence model pre-trained by de-noising corrupted text. Brown *et al.* [17] demonstrate the few-shot learning capabilities of GPT3, showing its ability to perform diverse tasks with minimal task-specific training data. The technical paper "GPT-4 technical report" by OpenAI [18] provides a comprehensive overview of GPT-4's development and capabilities. Lan *et al.* [19] introduce ALBERT, a parameter-reduction technique for BERT that maintains competitive performance while reducing model size. On the other hand, Xie *et al.* [20] propose a method for unsupervised data augmentation to improve the robustness and generalization of language models. Radford *et al.* [21] explore the use of natural language supervision to pre-train vision transformers, highlighting the potential of cross-modal learning. Lin *et al.* [22] survey de-biasing techniques for language models and evaluate their effectiveness on various benchmarks. Schölkopf *et al.* [23] discuss the importance of causality in representation learning and its implications for building more interpretable and reliable language models. Bender *et al.* [24] discuss ethical challenges related to NLP, including bias, fairness, and responsible AI development, which are relevant to LLMs. Sun *et al.* [25] review techniques for mitigating gender bias in NLP tasks, including those involving LLMs. Bakker *et al.* [26] present a method for fine-tuning language models using human feedback, addressing challenges related to controllability and alignment with user preferences. Ding *et al.* [27] discuss various aspects of LLMs, including their use in specialized domains and the challenges of understanding domain-specific terminology. Weiss *et al.* [28] discuss the various transfer learning techniques with case studies. Guo *et al.* [29] introduce DeepSeek-R1, which incorporates multi-stage training and cold-start data before reinforcement learning. This paper also demonstrates that the reasoning patterns of larger models can be distilled into smaller models, thereby reducing the requirements in terms of computing resources.

3. CONCERNS AND CHALLENGES

One major concern in LLMs is about the potential misuse of LLMs for generating harmful content such as misinformation, hate speech, or deep-fake text. A specific example of harmful misuse of an LLM involves the 2020 incident where a deepfake text generator was used to create a fake interview with a prominent political leader. Ensuring responsible use and mitigating harmful applications is a significant challenge. LLMs can inadvertently perpetuate or amplify biases present in the data they are trained on. This bias can manifest in various forms such as gender, racial, or cultural biases, leading to unfair or discriminatory outputs [30]. A specific example of LLMs perpetuating bias occurred with a model used for recruitment and hiring, where the AI was trained on historical data of past hiring decisions, which were already influenced by bias. In this case, the model exhibited gender bias, favoring male candidates over female candidates for technical roles, even though both genders had similar qualifications. LLMs can be exploited to generate malicious content, such as phishing emails, fake news articles, or even malware. LLMs can infringe on user privacy, especially in cases where they are trained on sensitive or personal data. There are concerns about the privacy implications of generating text that may inadvertently reveal confidential information or compromise user privacy [31]. In one case, users noticed that when asked about private details, like personal medical histories or conversations that were shared with AI models in earlier versions, the model sometimes produced outputs that seemed to recall specific details-information that was never directly provided in the query. Training LLMs requires significant computational resources, which can have a considerable environmental impact in terms of energy consumption and carbon emissions. Understanding and interpreting the outputs of LLMs can be challenging due to their complexity and lack of transparency. LLMs also lack in explainability. For instance, while the model may provide a rejection decision, it doesn't offer clear reasoning behind why one loan applicant is approved and another is denied, especially if both applicants have similar financial profiles. Finally, LLMs may struggle with understanding domain-specific knowledge or context, leading to inaccuracies or irrelevant outputs in certain domains. The details of this last challenge are explained in the subsequent sections.

4. DOMAIN-SPECIFIC KNOWLEDGE AND CONTEXT

Domain-specific knowledge and context pose significant challenges for LLMs due to their generalist nature. In one case, an LLM was used in a clinical setting to suggest treatments for a patient with a rare autoimmune disease, but it recommended a standard treatment for more common conditions, ignoring the nuanced, evidence-based protocols required for that specific disorder. These challenges derive from several

major factors, including lack of domain expertise, understanding specialized terminology, contextual understanding, data bias, and transfer learning limitations. Some sample outputs from popular LLMs such as ChatGPT 4.0, Sonnet, BERT, and Llama are presented in Table 1. These factors and some possible solutions are discussed next.

Table 1. Sample outputs from popular LLMs showcasing domain-specific challenges

| LLM_Name | Output | Domain | Challenge |
|------------|--|-------------------|--|
| PT-4o | The common treatment for neurofibromatosis includes using penicillin. | Medical | Incorrect recommendation: Penicillin is not a treatment for neurofibromatosis, highlighting the model's lack of domain-specific medical expertise. |
| BERT | The character sherlock holmes first appeared in the book the hound of the baskervilles. | Literature | Misleading: Sherlock holmes first appeared in a study in scarlet, not the hound of the baskervilles. |
| Claude 3.5 | In the field of quantum mechanics, the Schrödinger's cat experiment was designed to show how an electron can be both alive and dead at the same time. | Physics | Incorrect explanation: Schrödinger's cat is a thought experiment related to quantum superposition, not electron states. |
| GPT-4 | In the movie The Phantom Horizon (1995), the lead role was played by John Doe. | Entertainment | Hallucination: The Phantom Horizon is not a real movie, and John Doe is not an actor associated with it. |
| PaLM 2 | The square root of 64 is 8, and the capital of Germany is Berlin, but the Eiffel Tower is in London. | Geography | Incorrect information: The Eiffel Tower is in Paris, not London, demonstrating a lack of geographical context. |
| LLaMA | The treatment for Type 2 Diabetes often involves insulin injections, despite the fact that it is usually managed through diet and oral medication. | Medical | Misleading: Insulin is primarily used for Type 1 Diabetes, not as a first-line treatment for Type 2. |
| Sonnet | To find a document related to 19 th -century literature, search for keywords like '19 th -century novels,' or 'Shakespeare's plays.' | Literature search | Overgeneralization: Shakespeare's works are from the 16th century, not the 19 th , showing a lack of context awareness. |
| DeepSeek | To find a document related to 19 th -century literature, search for keywords like '19 th century novels,' or 'Shakespeare's plays.' | Literature search | Overgeneralization: Shakespeare's works are from the 16th century, not the 19 th , showing a lack of context awareness. |

4.1. Lack of domain expertise

LLMs are trained on a variety of text, all of which are derived from the internet, and therefore, cover many topics. While this enables them to generate text on a wide array of subjects, it also means they lack in-depth expertise in any specific domain [32]. As a result, when faced with domain-specific queries or tasks, LLMs may produce inaccurate or irrelevant responses. In other words, the LLM model is unable to accurately understand, interpret, or generate text related to specialized fields. This leads to incorrect or oversimplified responses when dealing with complex, technical topics. For example, in medical diagnosis, given the query "What are the differential diagnoses for a patient presenting with jaundice, elevated liver enzymes, and dark urine?" The LLM responds with the answer "The differential diagnoses for jaundice could include liver disease, gallbladder problems, or maybe some kind of infection. You should see a doctor for a proper diagnosis." The response is very general (lacks specificity), does not mention differential diagnoses (omission of key diagnoses), and fails to explain why the specific symptoms point towards these conditions (no detailed understanding).

4.2. Understanding specialized terminology

Many domains have specialized terminology or jargon that may not be commonly used in everyday language. LLMs may not understand or correctly use such terminology without specific training or exposure to domain-specific corpora. This leads to misunderstandings or misinterpretations of input text, and LLMs then produce nonsensical or incorrect outputs. This is particularly challenging in fields like medicine, law, engineering, and finance, where precise language is crucial. For example, assume that the user query is "Can you explain the implications of the doctrine of stare decisis in common law systems?" The LLM responds with "Stare decisis is a legal principle that courts should follow previous decisions in similar cases. This helps ensure consistency and predictability in the law." The response provides a basic definition but lacks depth in explaining the broader implications and applications of the doctrine of stare decisis. Also, it does not cover the details such as the distinction between binding and persuasive precedents, or how this doctrine affects lower versus higher courts.

4.3. Contextual understanding

Effective communication often relies on understanding the context in which information is exchanged. LLMs may not grasp the specific context of domain-specific conversations, leading to misinterpretations or inappropriate responses. For example, they may fail to recognize subtle cues or references that are crucial for understanding the meaning of a text in a particular domain that appropriately considers the broader context of a conversation or document. This can lead to misunderstandings or irrelevant responses. For example, in the domain of customer support interaction, given a customer query that read as “Hi, I ordered a laptop last week and it arrived yesterday, but it's not turning on. What should I do?” The LLM responds with the statement “Thank you for your purchase! To place a new order, please visit our website and browse through our wide selection of laptops.” This response does not address the user's problem of the non-functional laptop. Instead, it provides information on how to place a new order, which is not helpful in this context. The response also does not offer any troubleshooting steps, return policy information, or customer support contact details, which are relevant to the user's issue.

4.4. Data bias

The data used to train LLMs may not adequately represent all domains equally. Certain domains or topics may be underrepresented or misrepresented in the training data, leading to biases in the model's understanding of those domains. This can result in skewed or inaccurate outputs when generating text related to those domains. A simple example for data bias is the domain of job application. When prompted for a template for a job application, the LLM assumes that the applicant is a male and returns a default male name and gender-specific words such as ‘he’.

4.5. Transfer learning limitations

Transfer learning leverages knowledge gained from pre-training on a large, diverse corpus of text data to enhance performance on specific downstream tasks or domains. Transfer learning allows LLMs to adapt to specific tasks or domains through fine-tuning. However, it does not fully address the challenges of domain-specific knowledge and context. For example, assume LLM is given the problem of document analysis with the source domain being “general language modeling and understanding of everyday English”, and the target domain being “analysis and summarization of legal documents”. Further, assume that the user requests for summarization of an excerpt of a legal document “The party of the first part agrees to indemnify and hold harmless the party of the second part against all liabilities, losses, damages, and expenses, including attorney's fees, which may arise or result from any breach of this Agreement or from the acts or omissions of the party of the first part, its agents, or employees.” The LLM responds as “The first party will protect the second party from any problems that arise” which is over-simplified, completely non-legal in nuance, and misinterpreted.

4.6. Hallucinations

Hallucination occurs when an LLM generates information that is incorrect, fabricated, or inconsistent with reality [33], often due to the model's inability to properly handle specialized knowledge or context. This can happen because the model has general knowledge but lacks a deep understanding of specific legal language, precedents, or contextual factors, leading to the generation of misleading or completely false information. For instance, if a user asks an LLM - “Who played the lead role in the 1995 film The Phantom Horizon?” the model might invent an actor and provide a detailed backstory, even though no such movie or actor exists.

5. SOLUTIONS

Some of the domain-specific challenges faced by LLMs are discussed in section 4. Many of these challenges can be mitigated by fine-tuning, human intervention, and the use of benchmarks, to name a few. These, and many other solutions, are presented in this section.

5.1. Solutions for lack of domain expertise

Several solutions have been proposed to address the lack of domain expertise in LLMs. Fine-tuning LLMs on domain-specific datasets can help them adapt to the vocabulary, style, and intricacies of a particular domain [34], [35]. By exposing the model to domain-specific examples during fine-tuning, it can learn to generate more accurate and contextually relevant text for that domain. Instead of starting from generic pre-training, LLMs can be pre-trained on domain-specific corpora or with domain-specific objectives. This allows the model to capture domain-specific knowledge and patterns during pre-training, leading to better performance in that domain. Knowledge distillation techniques are another solution for lack of domain

expertise, which involve transferring knowledge from domain experts to the LLMs. This can be done through supervised learning, where experts provide annotations or corrections to the model's outputs [36], or through interactive methods where experts guide the model's behavior in real-time.

Prompt engineering involves designing tailored prompts or input formats that guide the LLMs to produce domain-specific outputs. By providing context or constraints relevant to the domain, prompt engineering can help steer the model towards generating more accurate and relevant text. Data augmentation techniques can be used to increase the diversity and coverage of domain-specific training data [37]. Either synthesizing additional training examples or augmenting existing ones can enable LLMs better capture the characteristics of a particular domain. Ensemble methods combine multiple LLMs trained on different domains or using different pre-training strategies. By aggregating the outputs of diverse models, ensemble approaches can improve robustness and performance across a range of domains. Hybrid models combine the strengths of LLMs with domain-specific models or knowledge bases. By integrating domain-specific modules or external knowledge sources, hybrid models can leverage both the generalization capabilities of LLMs and the specialized expertise of domain-specific models. Zero-shot and few-shot learning techniques enable LLMs to perform tasks in new domains with limited or no training data. By leveraging transfer learning and meta-learning approaches, LLMs can generalize across domains and adapt to new tasks more effectively.

5.2. Solutions for understanding specialized terminology

Pre-training LLMs on domain-specific corpora ensures that the model is exposed to the terminology and context of a particular field [38]. For instance, training on medical literature for healthcare applications is a good example for pre-training on domain-specific corpora [39]. Continuing the pre-training phase with additional domain-specific data to enhance the model's understanding of specialized terms can also help understand specialized terminology [40]. Fine-tuning LLMs on datasets that are specifically curated for a particular domain or task can significantly improve their performance in understanding and generating relevant terminology. Using labeled data (supervised learning) where the correct usage of specialized terms is explicitly marked, helps the model learn the correct context and meaning. Integrating structured knowledge bases (Knowledge graph integration) like medical ontologies or legal databases can help LLMs access precise definitions and relationships between specialized terms. Linking entities in the text to their corresponding entries in a knowledge base ensures the model understands and uses the correct terminology. Directly integrating specialized glossaries and dictionaries into the model's training data helps in understanding and correctly using domain-specific terms. Allowing the model to dynamically access and query specialized glossaries during inference improves accuracy in real-time.

Expert annotations involving domain experts and possibly knowledge injection [41] to annotate and review model outputs ensures the correct usage of specialized terminology. Systems where experts can interactively correct and provide feedback to the model using knowledge graphs [42] enables the model to learn from these corrections. Continuously updating the model with new data and terminology as the domain evolves, ensures that it stays current with the latest terms and their meanings [43]. Implementing mechanisms for the model to learn and adapt to new terminology dynamically (adaptive learning) as it encounters them in new texts is also a good mechanism for understanding new terminology. Specialized model architectures such as hybrid models, which combine general LLMs with smaller, domain-specific models that are experts in understanding and generating specialized terminology, and modular approaches which use a modular architecture where different components of the model specialize in different domains and can be selectively activated based on the task are equally useful for understanding specialized terminology.

Mixture of experts (MoE) is a variation of the expert annotations solution that involves using multiple specialized models or experts within a larger model, each trained on different domains or tasks. When a user queries the model, it activates only the most relevant expert(s) for that particular domain [44] [45], enabling the model to access highly specialized knowledge without overloading the system. This approach allows LLMs to handle diverse domains more effectively, ensuring that complex tasks—such as medical diagnosis or legal advice—can be processed by the most appropriate expert. MoE helps mitigate the challenge of generalized knowledge, offering tailored, domain-specific responses by dynamically selecting the right "expert" for the task at hand. Developing and utilizing domain-specific benchmarks specifically designed to test the model's performance on understanding specialized terminology in various domains and conducting regular evaluations and updates ensure that the model maintains high performance in handling specialized terminology.

5.3. Solutions for lack of contextual understanding

Enhanced pre-training techniques such as special training models using high quality data [46] and model architectures can reduce lack of contextual understanding. Special training models process longer context windows, enabling them to understand and retain more information from previous parts of a

conversation or text. Special model architectures that inherently consider context, such as transformer models with enhanced attention mechanisms (that can better capture dependencies across longer text spans) help with contextual understanding. LLMs can be fine-tuned in two ways—on datasets that emphasize contextual coherence and continuity [47], such as long-form conversations, chain-of-thought [48], or narrative texts (context-rich fine-tuning), and using dialogue datasets where the context is critical for maintaining the flow and relevance of the conversation, helping models learn the intricacies of context-dependent interactions (conversational fine-tuning).

Another solution is the use of memory-augmented models such as external memory mechanisms which deals with integrating external memory components to allow models to store and retrieve relevant contextual information as needed during inference, and retrieval-augmented generation (RAG), which combines retrieval mechanisms with generation [49], where the model retrieves relevant documents or context pieces from a large corpus to aid in generating contextually accurate responses. Two hierarchical models are of importance in context understanding. The first is the implementation of hierarchical attention mechanisms that can process context at multiple levels (e.g., sentence, paragraph, and document) to maintain a coherent understanding over longer texts. The second is using models that can process and integrate context at different granularities, ensuring a comprehensive understanding of both local and global context, also called layered context understanding. Utilizing contextual embeddings that adapt based on the surrounding context, ensures that the meaning of words and phrases is accurately captured in varying contexts. The same can also be implemented by employing cross-attention mechanisms that can dynamically adjust the focus on relevant parts of the context during inference.

Designing context-aware loss functions that penalize incoherence and contextually irrelevant outputs encourages the model to generate more contextually appropriate responses. One could also implement sequential training objectives that explicitly focus on understanding and maintaining context across sequences, such as masked language modeling with context windows. A prompting framework such as LLM4CS can be integrated into LLMs for a more efficient determination of context [50]. Human-in-the-loop systems such as those that allow human users to provide real-time feedback on the model's outputs, help the model learn to better maintain and utilize context in future interactions. Along with this, leveraging expert annotations help to correct and guide the model in understanding complex contexts, improving its contextual comprehension over time.

Another solution is the development of contextual benchmarks specifically designed to test the model's ability to understand and generate contextually coherent outputs such as long-form QA datasets or multi-turn dialogue datasets. In addition to this, conducting regular evaluations ensure the model maintains high performance in understanding and utilizing context across diverse applications. Also, integrating external world knowledge sources (e.g., databases and ontologies) provide additional context that can help the model understand and generate more contextually appropriate responses. The same can also be obtained by implementing mechanisms that allow the model to dynamically retrieve and incorporate relevant world knowledge based on the context.

Finally, slow thinking, a concept used in models like OpenAI O1, can be implemented to give the model more time and resources to reason through complex problems (rather than relying on quick, generalized outputs). By slowing down its processing and engaging in a more deliberate, deeper reasoning, the model can better understand the intricacies and context of domain-specific queries. This approach improves accuracy by enabling the model to consider additional layers of information, cross-reference details, and reduce errors, especially in specialized areas that require a deeper understanding, such as law or scientific research.

5.4. Solutions for data bias

Curating datasets that are more balanced and representative of different demographics, cultures, and viewpoints helps reduce biases in training data. Similarly, one should use techniques to augment under-represented data points, ensuring that minority groups are adequately represented in the training process. Implementing tools and algorithms to detect biases in datasets before and after training can highlight biased patterns that need addressing [51]. Utilizing adversarial training methods where a secondary model (adversary) is trained to detect and mitigate bias in the primary model and applying specific algorithms designed to reduce bias, such as reweighting, resampling, or modifying loss functions to penalize biased outcomes can also mitigate bias [52], [53]. Applying bias correction filters or adjustments to the model's outputs can correct biased responses after generation. Re-ranking or modifying the outputs ensures that they meet fairness criteria before being presented to users. Expert review involving human experts (RLHF, discussed in section 5.6) can help review and correct biased outputs, besides providing valuable feedback that can be used to retrain and improve the model. Using diverse groups of annotators (crowd-sourced annotators) can provide a wide range of perspectives and help identify biases that may not be obvious to a single demographic.

Maintaining thorough documentation of the data sources, model training processes, and known limitations or biases of the model helps users understand the potential biases and make informed decisions. Conducting regular audits of models to assess and document biases, ensuring accountability and continuous improvement. Experts should develop and adhere to ethical guidelines and frameworks that prioritize fairness and equity. These frameworks can guide the data collection, model training, and deployment processes. Equally important is the task of establishing policies and best practices for reducing bias, such as ensuring diverse team compositions and stakeholder engagement in the model development process. Regularly monitoring the model's outputs for biases and updating the model as new biases are detected can involve periodic re-training with more balanced data. There should be mechanisms implemented for users to provide feedback on biased outputs, which can be then be used to improve the model continuously. Active learning techniques can be used, where the model prioritizes learning from user provided examples that highlight biased behavior.

The model can take inputs using experts from various fields, including sociology, ethics, and law, to gain a comprehensive understanding of bias and how to address it effectively. The design and development of LLMs should incorporate principles of inclusivity and accessibility, which can help mitigate bias. Algorithms can be developed for learning fair representations of data that reduce the impact of biased features. Techniques can be implemented that ensure the model's decisions would remain unchanged in a counterfactual world where sensitive attributes are altered, promoting fair treatment. Ensuring compliance with policies and regulations related to AI fairness and ethics, such as general data protection regulation (GDPR) or the AI act proposed by the European Union by establishing clear protocols for transparency and accountability, including the ability to audit and explain the model's decisions and its potential biases.

5.5. Solutions for transfer learning limitations

Pre-training LLMs on domain-specific datasets provide a strong foundation in the relevant context and terminology before fine-tuning on specific tasks. Combining general and domain-specific pre-training [54] phases balances broad language understanding with specialized knowledge. Utilizing few-shot learning approaches where the model is fine-tuned on a very small amount of task-specific data leverages its preexisting knowledge effectively. Implementing meta-learning algorithms helps train the model to quickly adapt to new tasks with minimal data by learning how to learn during the training phase. Continuously updating the model with new data from different tasks and domains keeps it current and versatile. Using techniques such as elastic weight consolidation (EWC) prevents the model from forgetting previously learned tasks when trained on new ones (catastrophic forgetting mitigation).

Training models to develop task-agnostic representations that capture fundamental aspects of language makes them more adaptable to a variety of tasks. Also, use of self-supervised learning techniques creates robust representations that require minimal adjustment when transferred to new tasks. Training models with conditional inputs that specify the task or domain enables the model to adjust its behavior based on the given context. On the other hand, training the model with diverse and context-rich data improves its ability to generalize across different scenarios. Learning efficiency can be further improved by implementing active learning strategies where the model can query for the most informative data. By incorporating mechanisms for human feedback to correct and guide the model's learning process, one can enhance its ability to adapt to new tasks. One can also develop and utilize benchmarks specifically designed to test the model's transfer learning capabilities across various domains and tasks. Regularly conducting assessments of the model's performance on different tasks and domains can help in identifying and addressing any transfer learning limitations.

The use of knowledge distillation techniques where a large, well-trained model (teacher) transfers its knowledge to a smaller, task-specific model (student) can help the student model to learn effectively from the teacher's knowledge. Also, utilizing soft targets (probability distributions) rather than hard targets (class labels), can guide the student model, thereby improving its ability to generalize. In addition, introducing auxiliary tasks during fine-tuning, such as sentence completion, masked language modeling, or paraphrase detection, enhances the model's robustness and adaptability to new tasks. Conduction of additional pre-training phases with tasks closely related to the target domain facilitates smoother transitions and better performance on the new tasks.

5.6. Solutions for hallucinations

Reduction of hallucinations in LLMs can be addressed through several approaches. Fine-tuning LLMs on high-quality, domain-specific data helps improve accuracy and reduce the chances of generating incorrect or fabricated information. Incorporating RLHF allows the model to align more closely with human expectations and factual correctness. RLHF is a training approach where an AI model learns by receiving feedback from humans on its outputs. Instead of relying solely on predefined datasets, RLHF incorporates human judgments to guide the model's learning process. Users evaluate the model's responses, providing

feedback on quality, accuracy, and relevance. The model then uses this feedback to improve its behavior, aligning more closely with human preferences and expectations [55]. This iterative process helps refine the model, reducing errors, improving alignment with human goals, and addressing issues like bias or hallucinations in generated outputs.

Integrating automated fact-checking systems into LLMs ensures that generated content is cross-checked with verified databases, which also helps in minimizing errors. Controlled output generation can limit the model’s creative freedom, thus preventing speculative or false information. Combining LLMs with RAG allows real-time access to trusted data sources, grounding the model’s responses in factual content. Enhancing explainability and transparency helps trace how outputs are generated, enabling developers to identify and correct hallucinations. Regular updates to the model’s training data ensure relevance, while human-in-the-loop systems in critical applications provide expert oversight, further reducing the risk of hallucinations in high-stakes domains like healthcare or law.

6. CONCLUSION

This paper discusses some of the problems faced by typical LLMs specific to domain-related queries, such as lack of domain expertise, understanding specialized terminology, contextual understanding, bias, transfer learning limitations, and hallucinations. The details of these challenges are presented along with specific instances from popular LLMs. Some solutions such as fine tuning, slow thinking, human feedback, MoE, knowledge distillation techniques, task-agnostic representations, curation of imbalanced datasets, and use of memory-augmented models are also discussed for mitigation of these challenges.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

| Name of Author | C | M | So | Va | Fo | I | R | D | O | E | Vi | Su | P | Fu |
|---------------------|---|---|----|----|----|---|---|---|---|---|----|----|---|----|
| Kiran Mayee Adavala | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | |
| Om Adavala | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | |

| | | |
|-------------------------------|---------------------------------------|------------------------------------|
| C : C onceptualization | I : I nterpretation | Vi : V isualization |
| M : M ethodology | R : R esources | Su : S upervision |
| So : S oftware | D : D ata Curation | P : P roject administration |
| Va : V alidation | O : Writing - O riginal Draft | Fu : F unding acquisition |
| Fo : F ormal analysis | E : Writing - Review & E ditng | |

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] A. Vaswani *et al.*, “Attention is all you need,” in *31st Conference on Neural Information Processing Systems (NIPS 2017)*, California, United States, 2017, pp. 111.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” *arXiv-Computer Science*, pp. 1-16, Oct. 2018.

[3] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving language understanding by generative pre-training,” *Open AI*, pp. 112, 2018.




[4] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language models are unsupervised multitask learners,” *OpenAI*, pp. 124, 2018.

- [5] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le, "XLNet: Generalized autoregressive pretraining for language understanding," in *33rd International Conference on Neural Information Processing Systems*, Vancouver, Canada, 2019, pp. 57535763.
- [6] Y. Liu *et al.*, "RoBERTa: A robustly optimized BERT approach," *arXiv-Computer Science*, pp. 113, 2019.
- [7] Y. Sun *et al.*, "ERNIE 2.0: A continual pre-training framework for language understanding," *AAAI Conference on Artificial Intelligence*, vol. 34, no. 5, pp. 89688975, 2020, doi: 10.1609/aaai.v34i05.6428.
- [8] N. S. Keskar, B. McCann, L. R. Varshney, C. Xiong, and R. Socher, "CTRL: A conditional transformer language model for controllable generation," *arXiv-Computer Science*, pp. 118, 2019.
- [9] C. Raffel *et al.*, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, pp. 167, 2020, doi: 10.1145/3454287.3454799.
- [10] M. A. K. Raiaan *et al.*, "A review on large language models: Architectures, applications, taxonomies, open issues and challenges," *IEEE Access*, vol. 12, pp. 2683926874, 2024, doi: 10.1109/ACCESS.2024.3365742.
- [11] Y. Ge *et al.*, "OpenAGI: when LLM meets domain experts," in *37th Conference on Neural Information Processing Systems*, 2023, pp. 130.
- [12] D. Huang *et al.*, "DSQA-LLM: Domain-specific intelligent question answering based on large language model," in *AI-Generated Content*, Singapore: Springer, 2024, pp. 170180. doi: 10.1007/978-981-99-7587-7_14.
- [13] C. Di Sipio, R. Rubel, J. Di Rocco, D. Di Ruscio, and L. Iovino, "On the use of LLMs to support the development of domain-specific modeling languages," in *Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems*, New York, United States: ACM, 2024, pp. 596601. doi: 10.1145/3652620.3687808.
- [14] A. Holtzman, J. Buys, L. Du, M. Forbes, Y. Choi, and P. G. Allen, "The curious case of neural text degeneration," in *International Conference on Learning Representations*, 2019, pp. 116.
- [15] A. Bodor, M. Hnida, and N. Daoudi, "Integration of web scraping, fine-tuning, and data enrichment in a continuous monitoring context via large language model operations," *International Journal of Electrical and Computer Engineering*, vol. 15, no. 1, pp. 10271037, 2025, doi: 10.11591/ijece.v15i1.pp1027-1037.
- [16] M. Lewis *et al.*, "BART: Denoising sequence-to-sequence pretraining for natural language generation, translation, and comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Stroudsburg, United States: Association for Computational Linguistics, 2020, pp. 78717880. doi: 10.18653/v1/2020.acl-main.703.
- [17] T. B. Brown *et al.*, "Language models are few-shot learners," in *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, Vancouver, Canada, 2020, pp. 125.
- [18] OpenAI *et al.*, "GPT-4 technical report," *Open AI*, pp. 1-100, 2023.
- [19] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A lite BERT for self-supervised learning of language representations," in *8th International Conference on Learning Representations, ICLR 2020*, 2020, pp. 117.
- [20] Q. Xie, Z. Dai, E. Hovy, M.-T. Luong, and Q. V. Le, "Unsupervised data augmentation for consistency training," in *34th Conference on Neural Information Processing Systems*, Vancouver, Canada, 2020, pp. 113.
- [21] A. Radford *et al.*, "Learning transferable visual models from natural language supervision," in *38th International Conference on Machine Learning*, 2021, pp. 116.
- [22] Z. Lin, S. Guan, W. Zhang, H. Zhang, Y. Li, and H. Zhang, "Towards trustworthy LLMs: a review on debiasing and dehallucinating in large language models," *Artificial Intelligence Review*, vol. 57, no. 9, p. 243, Aug. 2024, doi: 10.1007/s10462-024-10896-y.
- [23] B. Schölkopf *et al.*, "Towards causal representation learning," *Proceedings of the IEEE*, vol. 109, no. 5, pp. 612634, 2021, doi: 10.1109/JPROC.2021.3058954.
- [24] E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell, "On the dangers of stochastic parrots: Can language models be too big?," *FACt 2021 - Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pp. 610623, 2021, doi: 10.1145/3442188.3445922.
- [25] T. Sun *et al.*, "Mitigating gender bias in natural language processing: literature review," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2019, pp. 16301640. doi: 10.18653/v1/P19-1159.
- [26] M. A. Bakker *et al.*, "Fine-tuning language models to find agreement among humans with diverse preferences," in *36th Conference on Neural Information Processing Systems*, 2022, pp. 114.
- [27] Q. Ding, D. Ding, Y. Wang, C. Guan, and B. Ding, "Unraveling the landscape of large language models: a systematic review and future perspectives," *Journal of Electronic Business & Digital Economics*, vol. 3, no. 1, pp. 319, 2024, doi: 10.1108/jebde-08-2023-0015.
- [28] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "Transfer learning techniques," in *Big Data Technologies and Applications*, Cham: Springer International Publishing, 2016, pp. 5399. doi: 10.1007/978-3-319-44550-2_3.
- [29] D. Guo *et al.*, "DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning," *arXiv-Computer Science*, pp. 122, 2025.
- [30] R. Patil and V. Gudivada, "A review of current trends, techniques, and challenges in large language models (LLMs)," *Applied Sciences*, vol. 14, no. 5, 2024, doi: 10.3390/app14052074.
- [31] I. Ullah *et al.*, "Privacy preserving large language models: ChatGPT case study based vision and framework," *IET Blockchain*, vol. 4, no. S1, pp. 706724, 2024, doi: 10.1049/blc2.12091.
- [32] P. Kumar, "Large language models (LLMs): survey, technical frameworks, and future challenges," *Artificial Intelligence Review*, vol. 57, no. 10, 2024, doi: 10.1007/s10462-024-10888-y.
- [33] B. Irfan, S.-M. Kuoppamäki, A. Hosseini, and G. Skantze, "Between reality and delusion: challenges of applying large language models to companion robots for open-domain dialogues with older adults," *Autonomous Robots*, vol. 49, no. 1, 2025, doi: 10.1007/s10514-025-10190-y.
- [34] T. Susnjak, P. Hwang, N. H. Reyes, A. L. C. Barczak, T. R. McIntosh, and S. Ranathunga, "Automating research synthesis with domain-specific large language model fine-tuning," *arXiv-Computer Science*, pp. 128, 2024, doi: 10.1145/3715964.
- [35] W. Zhao, H. Fan, S. X. Hu, B. Chen, and N. D. Lane, "CLUES: collaborative private-domain high-quality data selection for LLMs via training dynamics," in *38th Conference on Neural Information Processing Systems*, 2024, pp. 125.
- [36] Z. Ma *et al.*, "LLaMoCo: instruction tuning of large language models for optimization code generation," *arXiv-Mathematics*, pp. 121, 2024.
- [37] J. Yao, W. Xu, J. Lian, X. Wang, X. Yi, and X. Xie, "knowledge plugins: enhancing large language models for domain-specific recommendations," *arXiv-Computer Science*, pp. 114, 2023.
- [38] A. M. Bran, S. Cox, O. Schilter, C. Baldassari, A. D. White, and P. Schwaller, "Augmenting large language models with chemistry tools," *Nature Machine Intelligence*, vol. 6, no. 5, pp. 525535, 2024, doi: 10.1038/s42256-024-00832-8.
- [39] Y. Gu *et al.*, "Distilling large language models for biomedical knowledge extraction: a case study on adverse drug events," *arXiv-Computer Science*, pp. 115, 2023.




- [40] S. Pal, M. Bhattacharya, S.-S. Lee, and C. Chakraborty, "A domain-specific next-generation large language model (LLM) or ChatGPT is required for biomedical engineering and research," *Annals of Biomedical Engineering*, vol. 52, no. 3, pp. 451454, 2024, doi: 10.1007/s10439-023-03306-x.
- [41] R. Capellini, F. Atienza, and M. Sconfield, "Knowledge accuracy and reducing hallucinations in LLMs via dynamic domain knowledge injection," *Research Square*, pp. 18, 2024, doi: 10.21203/rs.3.rs-4540506/v1.
- [42] N. Ibrahim, S. Aboulela, A. Ibrahim, and R. Kashef, "A survey on augmenting knowledge graphs (KGs) with large language models (LLMs): models, evaluation metrics, benchmarks, and challenges," *Discover Artificial Intelligence*, vol. 4, no. 1, 2024, doi: 10.1007/s44163-024-00175-8.
- [43] R.-S. Lu, C.-C. Lin, and H.-Y. Tsao, "Empowering large language models to leverage domain-specific knowledge in e-learning," *Applied Sciences*, vol. 14, no. 12, 2024, doi: 10.3390/app14125264.
- [44] D. Chiba, H. Nakano, and T. Koide, "DomainLynx: Advancing LLM techniques for robust domain squatting detection," *IEEE Access*, vol. 13, pp. 2991429931, 2025, doi: 10.1109/ACCESS.2025.3542036.
- [45] X. Chen *et al.*, "Evaluating and enhancing large language models' performance in domain-specific medicine: explainable LLM with DocOA," *Journal of Medical Internet Research*, vol. 26, 2024, doi: 10.2196/58158.
- [46] Z. Zhao, E. Monti, J. Lehmann, and H. Assem, "Enhancing contextual understanding in large language models through contrastive decoding," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, Stroudsburg, United States: Association for Computational Linguistics, 2024, pp. 42254237. doi: 10.18653/v1/2024.naacl-long.237.
- [47] G. Marvin, N. Hellen, D. Jjingo, and J. Nakatumba-Nabende, "Prompt engineering in large language models," in *Data Intelligence and Cognitive Informatics*, Singapore: Springer, 2024, pp. 387402. doi: 10.1007/978-981-99-7962-2_30.
- [48] S. Ott *et al.*, "ThoughtSource: A central hub for large language model reasoning data," *Scientific Data*, vol. 10, no. 1, 2023, doi: 10.1038/s41597-023-02433-3.
- [49] M. H. Prince *et al.*, "Opportunities for retrieval and tool augmented large language models in scientific facilities," *npj Computational Materials*, vol. 10, no. 1, 2024, doi: 10.1038/s41524-024-01423-2.
- [50] K. Mao, Z. Dou, F. Mo, J. Hou, H. Chen, and H. Qian, "Large language models know your contextual search intent: A prompting framework for conversational search," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, Stroudsburg, United States: Association for Computational Linguistics, 2023, pp. 12111225. doi: 10.18653/v1/2023.findings-emnlp.86.
- [51] Y. Guo, Y. Yang, and A. Abbasi, "Auto-debias: Debiasing masked language models with automated biased prompts," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Stroudsburg, United States: Association for Computational Linguistics, 2022, pp. 10121023. doi: 10.18653/v1/2022.acl-long.72.
- [52] P. P. Liang, C. Wu, L. P. Morency, and R. Salakhutdinov, "Towards understanding and mitigating social biases in language models," in *Proceedings of Machine Learning Research*, 2021, pp. 112.
- [53] R. E. O. Roxas and R. N. C. Recario, "Scientific landscape on opportunities and challenges of large language models and natural language processing," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 36, no. 1, pp. 252263, 2024, doi: 10.11591/ijeecs.v36.i1.pp252-263.
- [54] Y. Xie, K. Aggarwal, and A. Ahmad, "Efficient continual pre-training for building domain specific large language models," in *Findings of the Association for Computational Linguistics ACL 2024*, Bang: Association for Computational Linguistics, 2024, pp. 1018410201. doi: 10.18653/v1/2024.findings-acl.606.
- [55] V. Rawte *et al.*, "The troubling emergence of hallucination in large language models-An extensive definition, quantification, and prescriptive remediations," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, 2023, pp. 25412573. doi: 10.18653/v1/2023.emnlp-main.155.

BIOGRAPHIES OF AUTHORS



Kiran Mayee Adavala    holds a doctor of Computer Science and Engineering degree from International Institute of Information Technology, Hyderabad (IIITH), India in 2014. She is currently an Associate Professor at Department of Computer Science and Engineering (AI&ML) in Telangana, Kakatiya Institute of Technology and Science, Kakatiya University, Warangal, India. Her research includes natural language processing, image generation, machine learning, data mining, internet of things, optimization and AI-companions. She has published over 42 papers in international journals and conferences. She can be contacted at email: ak.csm@kitsw.ac.in.



Om Adavala    received the B.Tech. degree in Computer Science and Business Systems from Jawaharlal Nehru Technological University, Hyderabad. He is pursuing his M.Tech. in Applied Data Science and Artificial Intelligence at National Forensic Science University, Gujarat, India. His research interests are in the area of forensic data analytics and large language models. His current work is in the application of deep learning for forgery and deepfake detection. He can be contacted at email: omadavala@gmail.com.