Indian Journal of Information Sources and Services
ISSN: 2231-6094 (P) Vol.15, No.2, 2025, pp.442-467

DOI: https://doi.org/10.51983/ijiss-2025.IJISS.15.2.54

Vulnerabilities and Defenses: A Monograph on Comprehensive Analysis of Security Attacks on Large Language Models

P. Balakrishnan[1] and A. Anny Leema[2*]

[1]School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India
[2*]School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India
E-mail: [1]balakrishnan.p@vit.ac.in, [2]annyleema.a@vit.ac.in
ORCID: [1]https://orcid.org/0000-0002-2960-636X, [2]https://orcid.org/0000-0002-0704-2794
(Received 26 March 2025; Revised 27 April 2025; Accepted 23 May 2025; Available online 25 June 2025)

Abstract - **This research mainly focused on highly developed natural language processing capabilities, such as large language models (LLMs), which can generate code and power chatbots, among many other uses. Their growing use, though, has put them under many security risks. This work thoroughly investigates LLM vulnerabilities, including adversarial attacks, data poisoning, prompt injection, privacy leaking, and model exploitation via jailbreak. Though there is an increasing corpus of defensive tactics, most still have limited reach, potency, or adaptability. The paper lists ideas for the following studies and emphasizes the requirement for strong, generalizable, explainable security solutions. Creating uniform evaluation standards, adaptive defense mechanisms, more transparent models, automated threat detection, and frameworks for ethical integration are all part of the approach. Ensuring LLMs calls for a multidisciplinary strategy that strikes a compromise between responsible government and technology innovation.**

Keywords: **Large Language Models, LLM Security, Data Poisoning, Prompt Injection, Jailbreaking, Model Robustness, Explainability, Defense Mechanism, AI Governance**

I. INTRODUCTION

GPT, BERT, and other models are used in the large language model (LLM), which has an evolutionary approach to Natural language processing to demonstrate the extraordinary capacity to comprehend and manipulate human text. The three modules are used in law, finance, healthcare, and education. These are all attached through academics. It also examines diverse security threats followed by LLM encounters, including data (Devlin et al., 2019) poisoning, adversarial prompting, and membership inference attacks. These harmful attacks affected model integrity and confidentiality to raise privacy, ethical application, and societal confidence queries. A strong defense system helps protect the model against various harmful elements. This work also determines to measure the defense for current and development privacy, adversarial training, and input sanitization. This research study aims to learn about more security issues associated with language problems to create a reliable AI system closely related to examining technologies to fix them.

1.1 Overview of Large Language Models (LLMs)

Language enables humans to express themselves when engaging with robots' communication (Chernyavskiy et al., 2021). Generalized models are needed since machines are increasingly expected to manage challenging language tasks, including translation, summarizing, information retrieval, conversational exchanges, etc. Lately, language models have shown notable advances mostly related to transformers (Jaber et al., 2025), enhanced computational capacity, and the availability of large-scale training data. These advances have resulted in a radical change by allowing the construction of LLMs capable of approximating human-level performance on different tasks (y Arcas, 2022). Large Language Models (LLMs) are cutting-edge AI systems that can read, write, and discuss text in a way that makes sense (Karimov & Bobur, 2024) and can be used for many jobs. Natural language processing (NLP) developed historically from statistical to neural language modeling and subsequently from pre-trained language models (PLMs) to LLMs. While PLMs are taught in a self-supervised environment on a vast corpus of text (Peters et al., 2018) to acquire a generic representation shareable among several NLP tasks, traditional language modeling (LM) trains task-specific models in supervised environments. Once adjusted for subsequent tasks, PLMs outperform conventional language models (LMs). The larger PLMs bring additional performance advantages, which have led to the shifting of PLMs to LLMs by notably increasing model parameters (tens to hundreds of billions) (Iyer & Deshpande, 2024) and training datasets (many GBs and TBs) (Lewis et al., 2019). The literature has presented several LLMs (Narayan & Balasubramanian, 2024). Early work on LLMs, Fig 1 shows that T5 and mT5 used transfer learning until GPT-3 revealed LLMs are zero-shot transferable to downstream tasks without fine-tuning. LLMs give correct answers to task questions when given job descriptions and examples (Xue et al., 2020). Nevertheless, and perform worse in zero-shot environments than in few-shot ones. Adjusting them with data from task directions (Zhang et al., 2021) and human preferences makes them better at generalizing to new tasks, significantly improving zero-shot performance and lowering misaligned behavior. Apart from enhanced generalization and domain adaptability, LLMs seem to have

emergent capacities in reasoning, planning, decision-making, in-context learning, answering in zero-shot environments, etc. Their grand scale makes them known to acquire these skills even if the pre-trained LLMs are not expressly trained to have these qualities (Mehta & Malhotra, 2024). From multi-modal, robotics, tool manipulation, question answering, autonomous agents, etc., these abilities have helped LLMs be extensively embraced in many contexts. Some suggestions for changes have also been made in these areas, such as task-specific training or better prompting (Workshop et al., 2022). The LLMs' capacity to do various tasks with human-level performance comes at slow training and inference, significant hardware needs, and more extraordinary operational expenses (Donkor & Zhao, 2024). As a result, these requirements have limited their use and created chances to create better architectures (Chung et al., 2024) and training strategies. Parameter efficient tuning, pruning, quantization, knowledge distillation, and context length interpolation are some methods that have been studied extensively to boost LLM performance (Arvisais-Anhalt et al., 2024). The research literature has lately seen a significant flood of LLM-related contributions as LLMs succeed in many tasks (Covington & Vruwink, 2024).

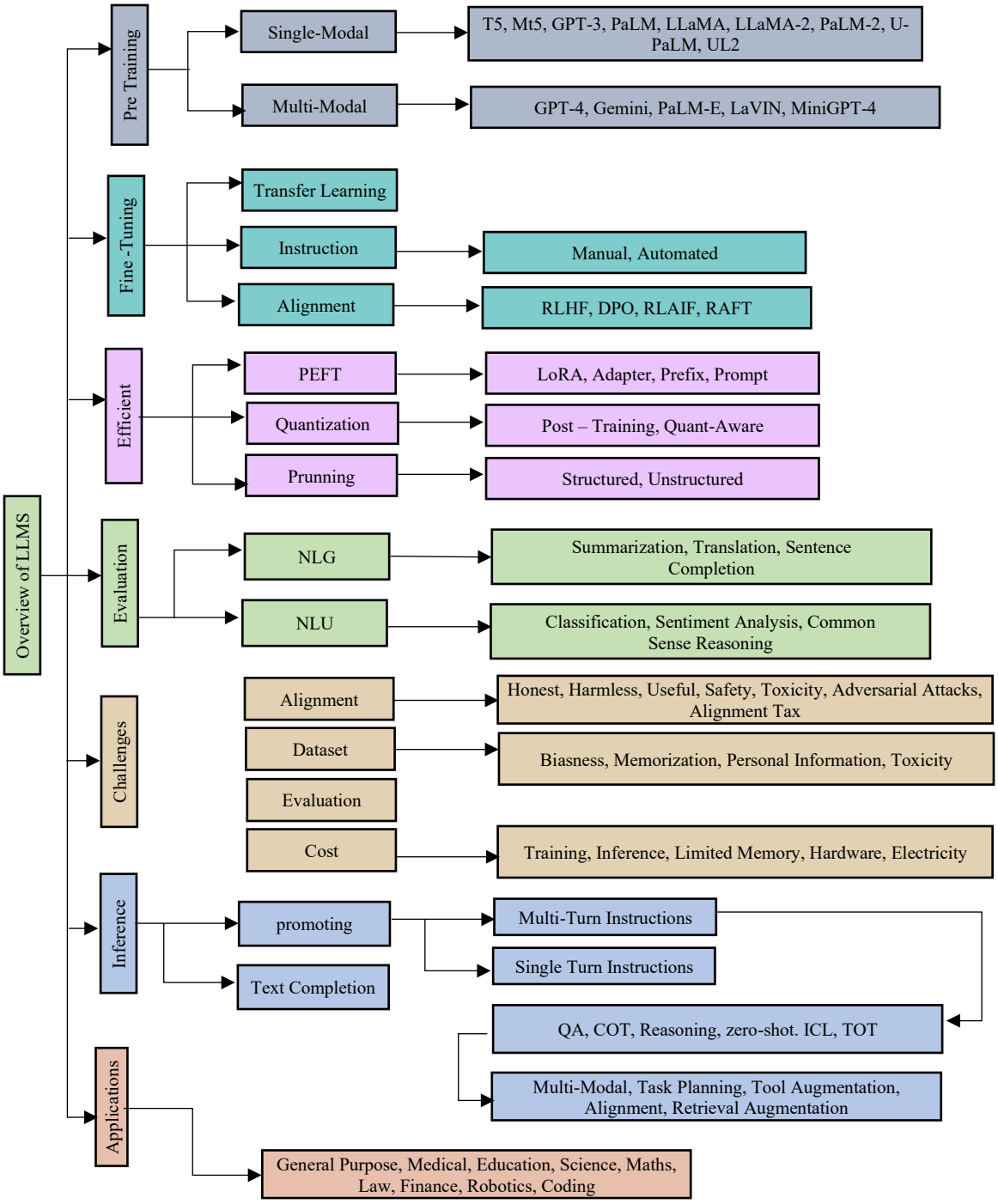

Fig. 1 Overview of LLMs

1.2 Importance and Applications of LLMs

LLMs are connected through transformative tools in AI due to various processes, massive datasets, and deep learning architectures. This model's importance as an LLM enables it to be connected to coherent and contextually related text. It's essential for NLP tasks. Scalability and Generalization, which means training with the traditional model followed by specific tasks such as LLM, should demonstrate the multiple domains with minimal fine-tuning. Democratization of AI and API, which have open-source platforms accessible through developers, researchers, and innovation experiments. Multimodal AI, LLM allows and integrates with the vision, speech, and robotic system to provide the way of holistic and multimodal AI applications (Krishnan & Patel, 2023). Here, we also explained the various types of key applications, such as customer support and chatbots, healthcare, education and E-Learning, legal and financial services, research, and data analysis. Here, it also explains the application of large language models in pathology as neural networks based on transformers, known as large language models (LLMs), that can produce responses to queries and commands that resemble humans. LLMs can produce instructional materials, summarize texts, extract structured data from unstructured text, write programs, create reports, and possibly help with case sign-out. When paired with vision models, LLMs can help understand histopathological images. LLMs have a vast potential to change pathology education and practice, but they are imperfect, so any AI-generated information must be checked with reliable sources. People should be careful about how these models are used in clinical settings because they can cause confusion and give wrong results, and relying too much on AI could lead to people losing their skills and machines being biased. This research article provides a short background of LLMs and discusses a few ways they can be used in pathology (Zhang et al., 2024). The following information describes the potential applications of LLM pathology and medicines. Medical report generation, a large language model, should be used to generate the model reports, followed by pathology and lab medicines. Generating these models is automatically followed to save time and reduce the workload for pathologists and lab professionals. Data analysis, LLM, is used to analyze the large volume of data followed by clinical data such as electronic health records to identify the patterns that help with disease diagnosis, treatment planning, and outcomes analysis. To analyze clinical decision support, LLM provides decision support through clinical applications, which includes patient education. LLM is used to develop chatbots or VR to get more information related to medical conditions through post-treatment care. In quality control, the LLM helps to identify the sources or quality issues, which helps to determine the accuracy and reliability. The medical literature analysis allows for extracting relevant information through clinical decision-making.

To analyze the benefits and drawbacks of LLM education, learning experiences are followed by LLM-enabled educational apps and provide effects related to the teaching and learning process. It's easy to understand it. Typically, analyze the trained large frequently used various types of the dataset, which is used for present difficulties producing the correct or pertinent results among the particular learning tasks. The investigating method applied through LLM addresses explicitly the various problems. It also understands the large amount of context combined with the pieces of information that many students have learned. Finally, analysis of the large amount of context connected through a lot of information promises to help students learn in the best way. To investigate the integration of an adaptive learning environment helps to develop more personalized, accurate, and proactive learning experiences.

The initial focus was mainly on open-source LLM for an automated educational process, followed by collecting feedback analysis and various prompting techniques. For example, it will examine sophisticated prompting techniques like chain-of-thought to assess how well different LLMs analyze educational survey responses, demonstrating that LLMs can perform at a level comparable to that of humans and expedite the examination of educational qualitative feedback (Raffel et al., 2020). Similarly, it shows the promise of GPT-4 by using few-shot learning to produce high-quality feedback that is better than expert human judgment in some situations, especially when giving trainee tutors remedial responses. Additionally, a few publications look at LLMs' potential uses, constraints, and strengths in automated assessment tasks. For instance, Morris, W., Holmes (2024) investigate how well LLMs, mainly GPT-3.5- generate evaluation replies in an undergraduate neuroscience course. According to reviews by graduate graders, LLMs perform on par with human students; nevertheless, these graders are more accurate than random guessing in differentiating between responses from LLMs and students (Henkel et al., 2024). Their experience with generative AI greatly influences the graders' detection accuracy. Experience with generative AI significantly impacts the graders' detection accuracy. Henkel et al. (2024) investigate how LLMs may be used to grade short-answer reading comprehension questions. They show that LLMs can attain grading accuracy that is on par with human experts, highlighting how LLMs can help in formative assessment grading (Morris et al., 2024).

The second element concentrates on pretraining and optimizing LLMs for specific learning contexts. Morris et al. (2024) present a mathematical automated scoring approach that can earn prizes. Their methodology includes extensive preprocessing to balance class labels, data augmentation for underrepresented classes, and customized input modification for various math issues. They attain nearly human-level accuracy on nine out of ten assessment problems by optimizing several pre-trained LLMs, such as DeBERTa, demonstrating the viability of scalable, reasonably priced solutions for grading extended response math questions. Zhang et al. (2024) created and tested a group of LLMs (like Llama and GPT-J) already taught with extensive and detailed math learning datasets for K–12 education. In various

downstream tasks, these models were compared to GPT-3.5, showing that focused pretraining can significantly improve LLM applicability in educational settings, sometimes surpassing GPT-3.5. (Acosta et al., 2024). To optimize the T5 model for automatically annotating epistemic and topic-related dialogue acts from students' chat exchanges, Acosta et al. (2024) present an original approach that integrates dual contrastive learning with label-aware data augmentation. The suggested approach outperforms the baseline model (such as BERT-based models), highlighting the value of augmentation strategies in improving LLM performance.

Enhancing adaptive digital learning environments through the use of LLMs is the primary objective of the third strand Norberg et al., (2024) use GPT-4 to update instructional math content to improve readability for implementation in an adaptive learning system. The researchers discovered that readability criteria like word frequency, sentence complexity, and semantic similarity could all be enhanced using GPT-4. Here, it also analyzes the impacts of K-12 student learning outcomes, which are followed by different subjects. Here, the analysis of some students is enhanced with the mathematical concepts that should be utilized in GPT-4 for revised problems (Chowdhary & Chowdhary, 2020). Based on these results, LLMs help with adaptive scaffolding and enhance the planning and goal setting in learning environments. To investigate the LLM in the educational sector, which continues improving the possible uses of various learning settings. The entire setting is followed by creativity and an analysis of the drawbacks, such as bias, which contains the possible outcomes. Thorough research in these fields can provide insightful information that can guide the ethical and successful incorporation of LLMs into various educational contexts (Adiwardana et al., 2020).

1.3 Motivation for Studying Security in LLMs

- It provides significant innovation through various sectors by analyzing the advancement technique and deployment of LLM, such as GPT, BERT, and other methods that directly interact with machines. These models help various applications, ranging from VR and automated content generation to personalized recommendations for multilingual translations.
- Adversaries exploit various systems through adversarial input, data poisoning, model inversion, evasion strategies, and membership inferences.
- It also ensures robustness and trustworthiness, which no longer exist. This paper aims to provide the gap and analysis of the security threats faced by LLM and explore the state-of-the-art defense mechanisms.
- Identifying, classifying, and analyzing the attacks contributes to a more resilient, ethical, and privacy-preserving AI system. We also collaborate with cross-disciplinary experts in machine learning, ethics, and policy, which shape the future and trustworthiness of LLM Deployment.

1.4 Structure of the Monograph

Various sections follow this monograph's research. Section I overviews LLM, its importance, and its applications. Section II describes the foundations of LLM, followed by the evolution of NLP, the architecture of GPT, BERT, and other models. It also defines the ethical considerations in LLM development. Section III describes security challenges in Machine learning, followed by an overview of vulnerabilities among machine learning systems and security risks. Section III describes various security challenges in machine language and consists of an overview of security in AI and Security risks to LLMs and adversarial machine learning. Section IV describes the types of security attacks in LLMs, followed by adversarial, poisoning, evasion, model inversion, and membership inference attacks. Section V describes defending against security threats followed by robustness, detection, defense mechanisms, various techniques, and privacy-preserving. Section VI explained real-world case studies. Section VII describes the ethical and legal implications. Section VIII discussed future directions in LLM security, and Section IX summarized key findings, future LLM security, and recommendations.

II. FOUNDATIONS OF LLMS

2.1 Evolution of Natural Language Processing (NLP)

The machine-learning technique, natural language processing (NLP), allows computers to understand, interpret, and modify human language. It is employed in several language-related tasks, including interacting with users, evaluating material, and responding to inquiries. Natural language generation, or NLG, and natural language understanding (NLU) are the two subfields of natural language processing (NLP). It is frequently combined with voice recognition software to convert spoken language into words. Sentiment analysis sorts texts by how they make you experience, while toxicity classification sorts texts by how unfriendly they are. Google Translate is one well-known example of machine translation, which is the automatic translation of one language into another. Entering entities in a text into established groups is called name entity recognition. This helps summarize news stories and fight fake news. In NLP, spam detection is a binary classification problem determining whether an email is spam. Grammar error correction stores rules for correcting grammar in text and is used by web grammar checkers and word processors. Latent Dirichlet Allocation (LDA) is a widely utilized method for assisting attorneys locate evidence in court papers. Finding the most important documents for a query is called information retrieval. Search and recommendation systems have to deal with this problem. Contemporary systems carry out two primary operations: indexing and matching (Johri et al., 2021).

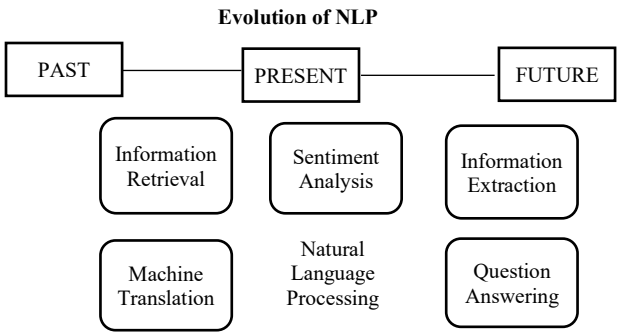

Fig. 2 Evolution of NLP

Natural Language Processing Fig. 2 involves significance from the rule-based system followed by the statistical model through deep learning and transformer architecture. It helps enable machines to understand and generate human language to measure fluency. After that, the Key NLP task consists of information retrieval, sentimental analysis, information extraction, machine translation, and Question answering. Information retrieval is an early stage of the key-based method to search the advanced semantic and contextual model followed by deep models. Which helps analyze conversational and cross-lingual systems. Sentiment analysis is grown through polarity detection followed by context-aware emotional interpretation with future directions. This sentimental analysis aims to analyze reasoning and personalization. Information extraction means shifting the pattern based on a system of neural networks followed by the complex entities among the relationships that move towards the open domain, multimodal extractions. The question-answering section is a temple-based method containing a large language model that includes open domain, multi-turn reasoning, and explainable QA systems. Machine translation is a rule-based statistical approach replaced by neural machine translations. In future advancement, we expect low-resource language support and cultural understanding (Vaswani et al., 2017).

2.2 Architecture of Large Language Models (GPT, BERT)

The large language model is an advanced type of deep learning model, which is trained for a large amount of text data to be understood and generate the human language. To build the architecture followed by GPT, BERT, T5. LLMs provide the contextual relationship between the sentences, phrases, and words. It should be performed by the various ranges of natural language processes, including translation, summarization, generation, sentiment analysis, and question and answer. LLMs should demonstrate remarkable generalization through achieving strong performances.

GPT Architecture

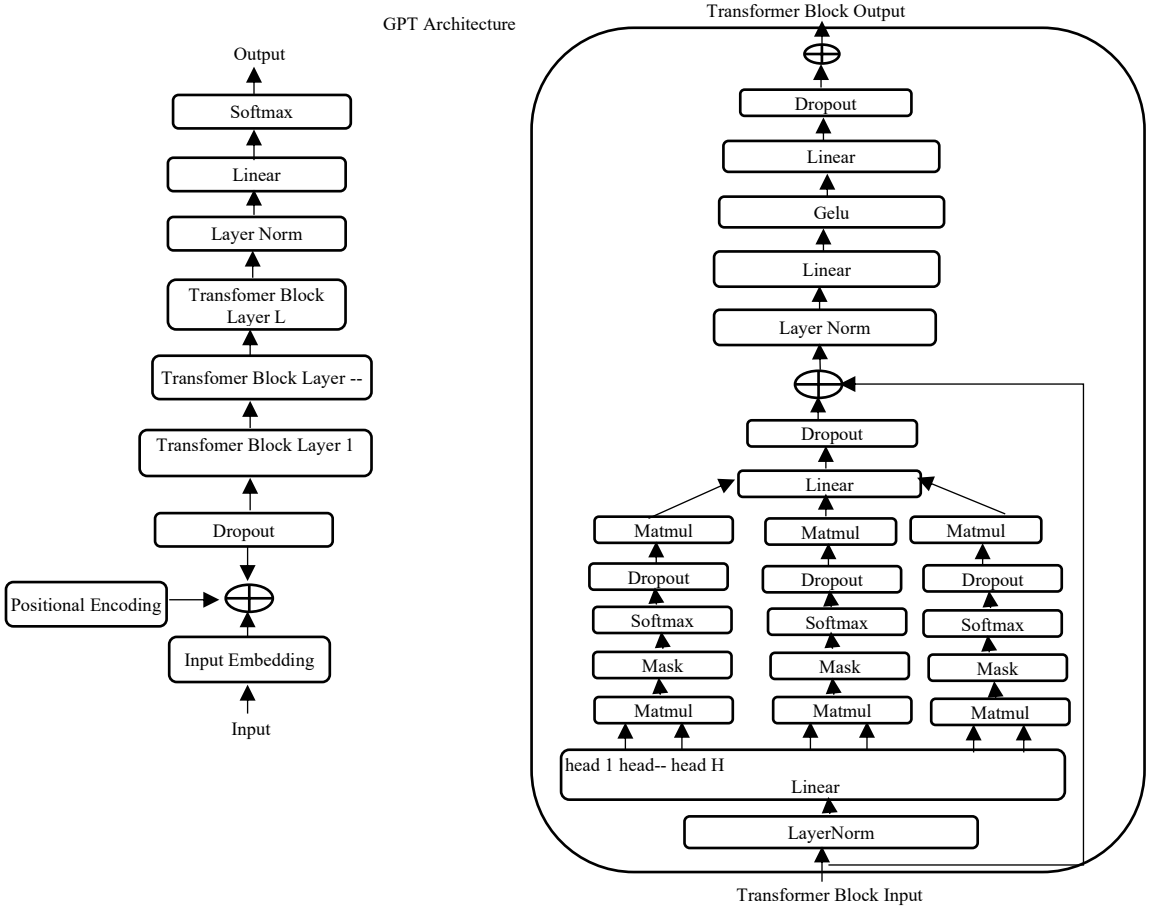

Fig. 3 GPT Architecture

Fig 3 explains Input Embedding, Positional Encoding, Dropout layers, Transformer Blocks, and layer stack. The Input Embedding contains the input and embedding; input is the raw text input tokenized through individual one. Embedding is a word that means each token should be converted into a dense vector that is represented by an embedding layer. Positional encoding, which means transformers are inherent, not directly, and positioned through encoding, should be added to input for the sequence of information. The dropout layer is used to prevent network output during the training period. The transformer block consists of layer norms, multi-head self-attention, add & norm, feed to forward network, and drop out. Layer stack means blocks are considered deeper models, allowing the network to capture more complex patterns and dependencies through input. The final Layer consists of layer form, linear, and softmax. The training process of Generative types of pre-trained transformers includes pre-training and fine-tuning. Pre-training, which is also known as language modeling. Fine-tuning, which means GPT models are performed as zero-shot and few-shot learning. It also improves the model performance and is specific to the domain.

BERT Architecture

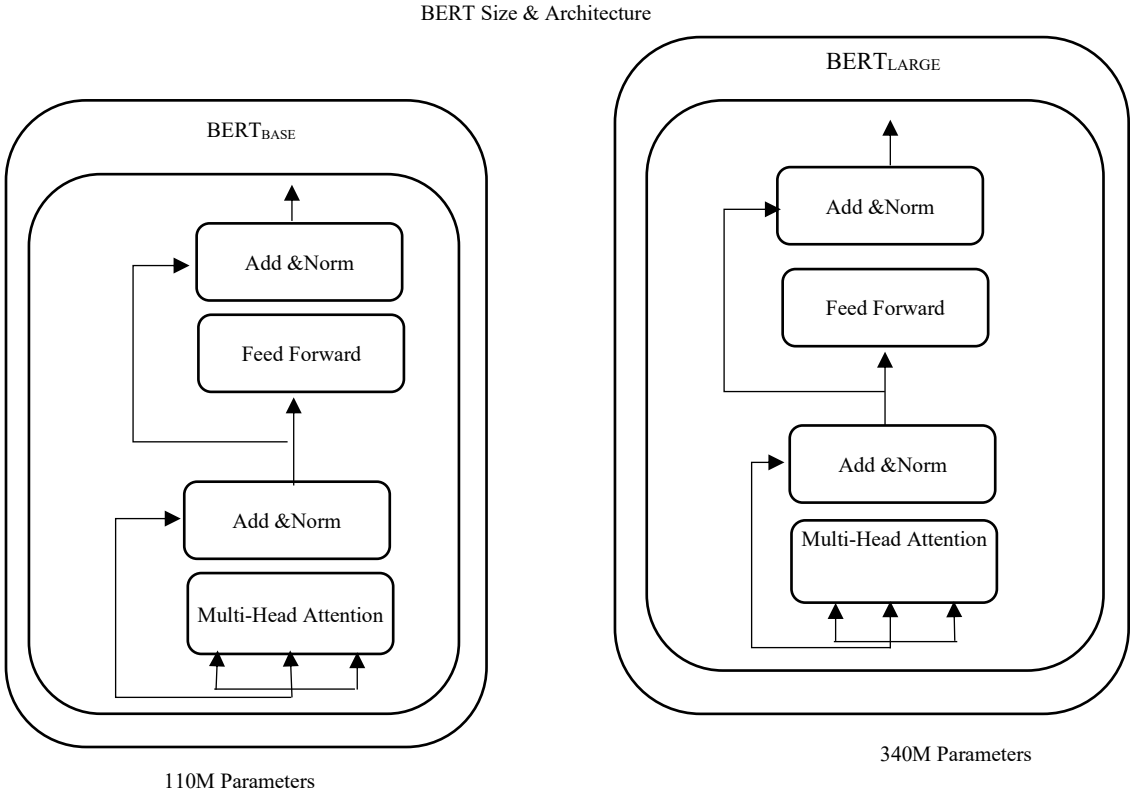

Fig. 4 BERT Size Architecture

The architecture Fig. 4 BERT NLP model, also called transformer architecture, is a sequence model consisting of an encoder and a decoder. To embed the input, we can use the tuned with embedded output as a string and consider it a decoder, similar to encoding and decoding algorithms. BERT architecture has a different structure, such as a traditional transformer. This model contains the stack encoder and others depending on the use cases. This also contains the CLS and SEP. CLS also represents the specific input classification. It is used for training models supervised through learning to understand the values. Consider the CLS value represented as 101. SEP is also represented as the unique token used as separate sentences. SEP token represented as 102. The applications of BERT NLP include sentiment analysis, language translation, Text summarization, matching and retrieving tests, google search, and Question answering. The advantages of this model are that it is pre-trained and contains more languages other than models. It helps to work as projects, not English ones. Comes under the BERT language model as fine-tuned with high accuracy. Architecture T5 Using a unified text-to-text training for all-natural language processing challenges, encoder-decoder model (T5). Layer normalization is put outside the residue path in a standard transformer model (Houlsby et al., 2019) by T5. Its pre-training goal is masked language modeling, whereby spans of consecutive tokens are replaced with a single mask rather than distinct masks for every token. This kind of masking generates shorter sequences, therefore accelerating the training. The model is refined using adaptive layers for downstream tasks following Table I pre-training (Brown et al., 2020).

TABLE I COMPARISONS FOR OTHER TYPES OF LARGER LANGUAGE MODELS

Ref No	Model Name	Description
(Hemasree & Kumar, 2022)	GPT-3	This model is based on the 175B parameter, with mechanisms for dense and sparse attention. It also demonstrates scale-driven performances.
(Mehta & Malhotra, 2024)	mT5	It's a multilingual version trained through T5 and contains the dataset as mC4 across the 101 languages. This model uses 250K vocabularies that fit specific languages. It allows through multi-lingual output generations.
(Workshop et al., 2022)	CPM-2	It is the English-Chinese model trained through WuDaoCorpus. It should be employed as the prompt tuning and memory efficient inferences.
(Kumar, 2024)	ERNIE 3.0	This model is called multi-tasking and is trained through various knowledge tasks.
(Wu et al., 2021)	Jurassic-1	It contains the J1 larger, which is used as balanced depth under the self-attention, flexible tokenization.
(Rae et al., 2021)	Yuan 1.0	Which is trained through Chinese text and used as MDFS, optimized through energy-efficient large-scale training with parallelism techniques.
(Bathala & Babu, 2024)	Gopher	This model contains a range of 44M to 280B parameters. The output performed value is GPT-3, Jurassic-1, and MT-NLG at 81%, which is mainly focused on scaling effects.

2.3 Training and Fine–Tuning LLMs

Pre-trained large language models consist of T5, mT5, and GPT-3, all providing a strong generalization among the required fine-tuning followed by the user values to avoid generating the contents. Which also improves the performance of the downstream tasks. Here, we also defined the fine-tuning strategies as adapter-based and prompt-based tuning, such as CPM-2 and HyperCLOVA. Modular architecture helps to optimize the different subsets of various model parameters for efficiency. Here are also the advantages of pre-trained models like GPT-3 and PanGU-α, which contain the multilingual model that should be performed competitively with the monolingual one (Lieber et al., 2021). Jurassic-1 also defined the architectural design choice containing token vocabularies and parallelism for better performances, specifically through various settings. Here, it also mentioned the overall thing as fine-tuning not only helps to improve safety but also helps to boost the generalization with minimal computations. Tuned large language models like T0, WebGPT, Tk-INSTRUCT, m10, BLOOMZ, OPT-IMI, Sparrow, and Flan Show. Which mainly describes diversity, human feedback, and prompt design. It also enhanced the zero-shot and generalization performances. After analysis, the model, To, and Flan Benefit from the multi-task prompt and CoT reasoned through better zero-shot learning. The other models, like WebGPT and Sparrow leverages, are human judgment and reinforcement learning. It also helps to improve factuality and dialogue safety. These models are more reliable, efficient, and adaptable. Instruction tuning with LLMs generated a dataset that often contains the input and output pairs to a smaller size and is less diverse. To overcome this issue, the proposed approach LLMs are used to generate the instruction tuning dataset. To train the self-instruct model, create the dataset as SUPER NATURAL INSTRUCTIONS by 33%.

2.4 Ethical Consideration in LLM Development

Privacy and Data Risks

In this age of electronic information technology, privacy concerns and data protection have always gone hand in hand with technological progress. LLMs have a lot of features and training data sets. They may "remember" sensitive personal information from their training data, which lets these models make information that includes specific personal data. The author verified the possibility of privacy leaking within the inference process shown by learning models. There is a chance that private information will get out when teachers or schools use data, like students' personal details, social background details, and health details, to train models to give students personalized feedback on their learning interests and progress. When learning management systems (LLMs) are utilized to facilitate learning, such as in clinical case discussions, tasks that involve sensitive patient data such as name, sex, age, medical history, or final diagnoses, and even image data such as computed tomography scans and magnetic resonance imaging to assist in interpretation, the confidentiality of patients is put at risk. Previous research indicates that some advanced models can reidentify such personal information from vast data sets using so-called linkage attacks, exposing information even if it is anonymized when it is entered. LLMs reportedly could reidentify 99.98% of personal data in any anonymized data set using as few as 15 demographic traits. Thus, identifying information alone is insufficient to safeguard patient privacy, which presents significant difficulties in the medical education framework when applying LLMs. In addition, since LLMs' training and user data are kept on cloud servers, any security holes could expose this sensitive information. Companies that develop LLMs, like OpenAI, may use users' personal data for service analysis, improvement, or research, and they can share it with other companies without consumer authorization. These elements raise privacy and data security threats in medical education systems.

AI Hallucination

Clinical practice guidelines and consensus agreements, which are constantly updated, are highly essential resources in medical education. Nevertheless, LLM "AI hallucinations" provide a significant challenge to properly leveraging these materials as instructional tools. LLMs generate responses that appear logical but are incorrect, inconsistent, or fabricated in medicine, including those that rely on fabricated data and falsified references, resulting in AI hallucinations. When the author asked ChatGPT 3.5 and Google BARD to respond to lung cancer-related queries, he found that these LLMs had inaccuracy rates ranging from 17.5% to 27.5%. Similar problems have been noted in dentistry. when LLMs provide erroneous, ambiguous, antiquated information. Research has

underlined accuracy issues regarding the reactions of LLMs in medical environments. Several elements in the world of medicine help to explain AI hallucinations. First, mainly from the internet, LLMs' training data comprises erroneous and out-of-date material lacking quality control. Second, access to authoritative databases such as PubMed, UpToDate, and Cochrane calls for subscriptions, limiting LLMs from getting current and trustworthy research data. Furthermore, AI hallucinations are made worse because LLMs can't always think and generate texts based on probabilities. In the medical field, most instructors employed with LLM-generated content help to transmit knowledge. Most medical students are related to teachers in various facts specifically acceptable through false information. Evaluating multiple students with LLM content should promote false information, which is mainly focused on improper assessment and ongoing education with knowledge. In most scholarly studies, LLM-generated information is extensively shared with medical education. This is why systematic mistakes in clinical decision-making compromise safety and health. Medical education also mentioned that public health in health treatment should generate ethical questions and problems about academic integrity, which helps to affect competence and social obligation through the medical field.

III. Security Challenges in Machine Learning

3.1 Overview of Security in AI and Machine Learning

In the prevention process, we use some attacks such as data poisoning, adversarial attacks, model inversion, and model theft; in security aspects, these attacks help fix flaws in the training data, model design, or deployment phase to change results. This is to ensure security calls in strong training mode, different levels of confidentiality, competitive defenses, and real-time monitoring. It also explains how security across the AI life cycle is integrated, including data collection, placement and monitoring, and dependability. These types of threats progressively affect the essential domains.

3.2 Vulnerabilities in Machine Learning Systems

Machine learning systems have their advantages in various phases of ML. Data poisoning means attackers insert false or manipulated information through a training set, leading to a compromised or biased model. Another harmful effect is usually the undetected changes through the input process, which produce various predictions. Most adversaries are extracted from the sensitive information related to training data used by the model inversion and membership inference attacks, which all have privacy implications. Furthermore, which is analysis, the various attackers are to copy the model and frequently analyze the deployed model. These issues are also discussed with strong security measures, including validating date data, training models as secure bases, and monitoring the deployment.

3.3 Security Risks Specific to LLMs

As with their design, capabilities, and practical implementation, models such as BERT and GPT raise unique security concerns. One of the primary issues is prompt injection malicious inputs that either take control of the prompt or evade the designed safeguard. Also, of concern is the potential for data leakage, which may occur when sensitive or proprietary information is inadvertently exposed during the training process of an LLM. The ethical and security concerns associated with model abuse include the creation of hazardous content, such as deep fakes and phishing emails intended to facilitate cyberattacks. LLM response manipulation can be achieved through training data poisoning, where biased or falsified information is introduced into publicly available datasets. Moreover, breaches of user privacy with membership inference and model inversion attacks grant adversaries details pertaining to the training data, compromising user privacy. These hazards underscore the importance of establishing safe model training, access control, output filtering, and monitoring on large language models.

3.4 Adversarial Machine Learning: A Threat to LLMs

Large language models, or LLMs as they are commonly abbreviated, are severely threatened by adversarial machine learning due to their focus on precisely crafted inputs. Models can easily be tricked into generating faulty, negative, or biased results through adversarial prompts or sequences of text. Such attacks may circumvent content filters, misappropriate sensitive data, or silently alter the model’s intended functionalities. An example of an adversarial attack is prompt injection, which involves embedding harmful directives within untrusted texts or user queries that textbook searches would overlook, allowing the attacker to commandeer an LLM’s logic. These attacks undermine the reliability and safety of LLMs, which are increasingly integrated into productivity applications, chatbots, and even search engines. Those threats pose severe risks to the credibility of the system. To counter such threats, robust input sanitization, continuous adversarial training, response filtering, and real-time monitoring to reduce detrimental interactions must be employed.

IV. Types of Security Attacks on LLMS

4.1 Adversarial Attacks

Adversarial attacks are machine learning techniques that intentionally manipulate the input devices or misleading techniques and analyze the vulnerabilities' exploit level to produce incorrect or unintended output. The author describes the context of LLM results as harmful and followed by bias. Unlike prompt hacking, which also manipulates the input prompts during inference, adversarial attacks should occur in training and inferences.

4.1.1 Definition and Techniques

Though intentionally followed by craft input, the definitions of adversarial attack in machine language are also called adversarial examples; it is one of the reasons the trained model makes incorrect predictions. This is followed by perturbations, which often provide imperceptibility through the human eye and exploit the weakness among the model decision boundaries. The adversarial attacks include Evasion attacks, Text perturbation attacks, poisoning attacks, Prompt injection, jailbreaking, data poisoning, and model inversion. Various techniques are used in adversarial attacks, such as Textual perturbation attacks consisting of character- and world-level attacks. Embedding Space attacks followed by Gradient-Based optimization and PWWS. Prompt Injection Attacks consist of Direct prompt injection and indirect prompt injections. The Backdoor or Trojan Attacks model helps to train the malicious triggers. Black box query attacks followed by transferability. Maintaining system integrity depends on identifying hostile assaults on LLMs. Essential strategies consist of methods like opposing steps, which involve preprocessing inputs to remove adversarial perturbations before they reach the model. Suggested adding dual-stance training, where the model is trained on untainted and attacked data to improve its ability to discern and denounce harmful inputs. This approach improves LLMs' resistance to malevolent examples. Robustness measures: introduced measures of how vulnerable a model is and when facing possible attacks. These measures help locate the exposed vulnerabilities in an LLM that adversaries can exploit and thus assist in reinforcing the prevent veiling model.

4.1.2 Impacts on LLMs

Adversarial attacks against Large Language Models (LLMs) seriously impact cybersecurity mechanisms, such as Compromised Threat Detection, encompassing Adversarial inputs that can potentially go unnoticed by LLM-based threat detection systems, enabling malicious actions to become undetectable. Degraded Decision-Making accounts for tampered model predictions caused by adversarial attacks that may lead to faulty security decisions, such as misidentifying benign behaviors as malicious or vice versa. Privacy violations, known as model inversion attacks, can harm users' privacy by allowing out sensitive or personal data generated by LLM. Recent Strategies for Adversarial Defense: Researchers have developed various methods to defend LLMs against adversarial attacks in cybersecurity environments. Adversarial Training is one of them, which assists in creating model robustness against adversarial modifications using purposely malicious examples. Removing potentially malicious inputs before passing them to LLMs is called input sanitization. This is accomplished through techniques such as input validation and anomaly detection. Reshaping model training goals to provide adversarial attack resilience is a priority, as is reducing worst-case loss or maximizing adversarial robustness metrics, which is referred to as robust optimization (Radiya-Dixit et al., 2021).

4.2 Poisoning Attacks

According to Wan et al.,2023), data poisoning attacks should be involved in celebrating the manipulation of the training data to compromise the AI model decision-making process. It also noted that some specific attacks are effective when training data is collected from external or unverified sources. It is easiest for the attackers to be introduced to the positioned examples dataset used in train language models.

4.2.1 Concept of Data Poisoning

Data poisoning intentionally corrupts a training set by using application programming interfaces to influence a machine learning model's learning process through deceptive or malicious data. It's a strong strategy since even little changes to a dataset can significantly affect a model's predictions and decision-making ability. An attacker can change the LLM's internal model of how language or code should work by slightly changing the statistical trends in the training data. This can cause results that are wrong or biased. Here is a recent actual case: Being available Attacks seek to ruin the general performance of the model. Attackers may tamper with labels (e.g., classifying a spam email as benign) or add loud or pointless material. Because of this, the model becomes less accurate and has trouble making accurate predictions. An attacker could exploit exposed API tokens with write permissions to introduce deceptive data to training sets, as demonstrated in the recent Hugging Face example. Targeting attacks seek to make the model misclassify a specific input type. For instance, an assailant might teach a facial recognition system to fail to identify a certain person by feeding it poisoned data. Perhaps the sneakiest kind of data poisoning, backdoor assaults, place secret triggers inside the model. An assailant might present standard visuals but with a precise pattern that, upon recognition later, will force the model to generate a desired but erroneous response. The injection is security flaws such as SQL injection or code injection in the API that can let the hacker control the entered data, which are described as flaws. Unencrypted data transfer between the API and data source could let hackers intercept and alter the training data.

4.2.2 Poisoning Attack Strategies

Various strategies exist to safeguard large language models (LLMs) against data poisoning attacks, and this area of research remains ongoing. Here are some salient features of essential strategies: Anomaly Discovery: Using BERT embedding distances to find examples that have been poisoned. Usually showing as anomalies in the data used for training distribution, poisoned data points might be filtered to help reduce their impact. Here, it also covers several methods, including eliminating known triggers, near-duplicate poisoned samples, and payloads capable of resisting TROJANPUZZLE attacks; organizing the data guarantees the elimination of abnormalities and dubious information. These techniques make defending LLMs against data poisoning attacks difficult. Empirical studies show that LLMs are progressively susceptible to such attacks; current

protections, such as data filtering or lowering model capacity, provide only limited protection at the expense of lowered test accuracy. Therefore, more effective defense methods are needed to balance model utility with the ability to protect LLMs from data poisoning attacks.

4.2.3 Case Studies of Poisoning Attacks

We also explained the case studies related to poisoning attacks in LLM, such as prompt injection through training data, which means directly focused LLM often to train the massive types of web scrapes such as a standard crawl. In this case, found by the attack, most of the attackers inserted misleading content through websites and forums and scraped training data. Real-world examples are considered fake Wikipedia pages pushing misinformation, and LLM is later repeated as fact. Impact as LLM should learn about harmful or biased through patterns and confidentially false (Pearce et al., 2025). Poisoning via code repositories, With the context of LLM, should be like Codex or CodeLLaMA, which is trained through open-source code. In this case, the attack was caused by malicious actors uploading the various repositories as insecure or misleading code patterns. The impacts of this case suggest that insecure coding practices should include backdoors to assist developers (Paracha et al., 2024).

4.3 Evasion Attacks

4.3.1 Methods for Evasion

Three adversarial evasion attack methods were chosen due to their distinct impacts on the robustness of the models, which are a result of their distinct approaches to generating adversarial examples. Because the BERT Attack uses its own BERT model to create word-level perturbations and target another LLM, it essentially puts two LLMs in competition. Because it uses a limited data generation method to substitute words with other appropriate possibilities in a preset checklist, the Checklist Attack was also used, even though it accomplishes identical word-level perturbations. The Typo Attack differs significantly in using more particular character-level perturbations, such as adding, removing, or switching adjacent characters within a word or characters for adjacent keys on a typical QWERTY keyboard. A rate of misclassification (MR) is the standard evaluation tool used to assess and compare the efficacy of adversarial attacks. It calculates the percentage of incorrectly classified text samples, indicating how well an attack fools a model. A low MR suggests that an LLM is more resilient to a particular attack since its value rises from 0% to 100% when more hostile instances are incorrectly classified.

The key elements of an attack's methodology, which are typically overlooked in straightforward performance reviews, must be considered to assess its effectiveness and viability. Two measurements were made to compare how many changes each attack method made and how much it cost to run: Average Perturbed Words (APW) and Average Required Queries (ARQ). The APW estimates how many changed words are in a given text sample by averaging all of the samples. Character-level changes didn't replace whole words, but these words were still considered disturbed because they were changed. The ARQ finds the average of all perturbed text samples based on how often a model's class estimates were questioned until the right perturbations were found and the text sample was wrongly classified. Through adversarial evasion attempts against each LLM, the efficacy, efficiency, and viability of each approach were assessed. As the attacks progressed, the metrics under consideration MR, APW, and ARQ were calculated, and the outcomes of each technique were examined and contrasted. The acquired results are summarized in the table II.

TABLE II MODELS RELATED TO VARIOUS ATTACKS

Model	Attack	MR (%)	APW (#)	ARQ (#)
BERT	BERT Attack	100	3.09	135.17
	Checklist Attack	40	1.06	2.17
	Typo Attack	50	1.58	381.1
RoBERTa	BERT Attack	80	3.14	104.5
	Checklist Attack	20	1.15	2.25
	Typo Attack	50	1.82	356.62
DistilBERT	BERT Attack	100	3.08	104.86
	Checklist Attack	30	1.12	2.1
	Typo Attack	50	1.64	355.2
ALBERT	BERT Attack	90	2.48	98.67
	Checklist Attack	10	1.09	2.11
	Typo Attack	80	1.7	339.15
XLNet	BERT Attack	100	2.43	85.33
	Checklist Attack	10	1.08	2.2
	Typo Attack	90	1.62	362.89

To compare the above, table II represents the five NLP models such as “BERT, RoBERTa, DistilBERT, ALBERT, and XLNET”, which are against adversarial attacks such as BERT attacks, checklist attacks, typo Attacks used by the various types of metrics such as MR (Misclassification Rate), APW (Average perturbations per word), ARQ (Average Replaced Queries. After evaluating this analysis, BERT shows the highest effectiveness among the High MR, up to 100% related to APW. Which is indicated as small; it is a contextual analysis of the heavy impacts of model predictions. Checklist attack results are lower than MR compared to the model at 10-40%, Suggesting rule-based perturbations. Typo attacks are constantly noted as moderate and highly effective, which is also required for many attempts. To analyze the overall model as effective.

4.3.2 Evasion in LLM Context

In Large Language Models (LLMs), evasion intentionally manipulates input cues to circumvent built-in safety systems and content filters. Attackers use obfuscation methods (e.g., substituting symbols for letters), synonym substitution, encoding techniques, or contextual masking to fool the model into producing limited or dangerous content and hide destructive intent. A malevolent actor could, for example, disguise a forbidden request as a hypothetical situation or employ code-like formatting to avoid detection. The production of damaging, prejudiced, or illegal content, eroding trust in AI systems, and enabling actual-world misuses like phishing or false information are likely results of these evasion tactics, which pose a serious threat.

Discovery and halting evasion are challenging because such inputs typically appear harmless at first look. Developers, therefore, need to implement sophisticated rapid filtering, adversarial training, context-sensitive moderation, and ongoing surveillance to ensure responsible and secure LLM deployment.

4.4 Model Inversion Attacks

4.4.1 Understanding Model Inversion

The model inversion attack is also hostile to reconstructing the user data as private from the output of the machine learning model. This approach has different implications for data security and privacy across the other applications. Model inversion attack is also mentioned as focused on reconstructing the private data of the user from the output of the machine learning model. It should be obtained, and the input data should be controlled to exploit the model's vulnerabilities. The primary aim of this attack is to receive several types of sensitive user information, including location, identity, and behavior. Model inversion attack followed by several steps to get the machine learning model as input. It's utilized to control the input data to receive the desired output. The model output is utilized to build the user's personal data. These attackers employ different techniques, like input data tampering, which is addressed by not limited sources, noise addition, and input perturbations, which are utilized as adversarial examples. The primary intention behind this attack is to build the input that must to yield the desired output of the model to be able to rebuild the private data of the user.

Different model inversion attacks are employed, including membership inference attacks, attribute inference attacks, and reconstruction attacks. Membership inference attacks identify the user data that should be employed to train the machine learning model. Attribute Inference attacks make inferences about the sensitive attributes of a user either based on age or location. Reconstruction attacks reconstruct the user's private data machine learning model, which is regarded as the model output. These attacks have features, which are accompanied by other challenges and uses. The main objective of this attack is to distrust and analyze the privacy of machine learning. This type of attack tries to get the adversary to steal and develop an ML model by replicating the understanding behavior and query with a different dataset. An adversary should extract the baseline model as it should represent the model inversion attack to regenerate the model's training data. According to that, this type of framework used as the model inversion attack, which contains the collaborative types of machine learning model, demonstrates the successful one. It also highlights the various impacts of model inversion attacks on transferring learning models. The below Fig. 5 should represent the model inversion attacks on machine learning (Maramreddy & Muppavaram, 2024).

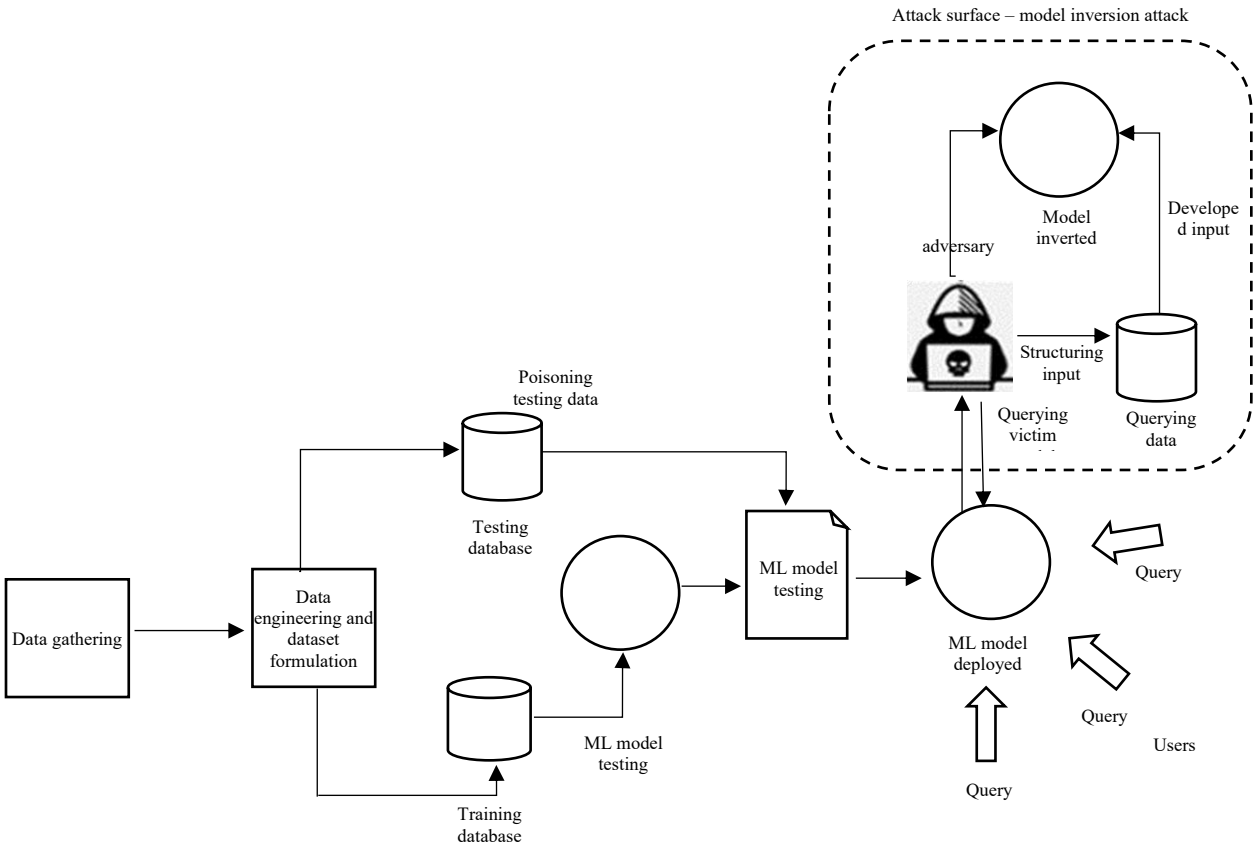

Fig. 5 Model Inversion Attack

4.4.2 Repercussions for Privacy

Based on the privacy concerns discussed, various factors include data leakage, compliance issues, ethical concerns, reconstructing personal data, and exposing confidential information. Data Leakage: If training data reconstruction is successful, private information may be revealed, privacy laws may be broken, and people or organizations may suffer harm. Compliance Issues: The potential for private data extraction raises questions regarding adherence to privacy laws such as HIPAA and GDPR. The ethical implications of exposing private information or data via model inversion assaults are significant, as they can potentially undermine faith in AI systems. Reconstructing Personal Data: If an LLM is trained on a dataset that contains private emails, phone numbers, or other personal information, an attacker might use that information to rebuild the data. Disclosing Private Information: An attacker might use an LLM trained on medical data to recreate patient records or private medical information. Intellectual Property Disclosure: An attacker might use an LLM trained on company-specific data to reconstruct trade secrets or sensitive information. Various privacy assaults exist, including model stealing and membership inference attacks. We thus quickly show these attacks and elucidate their variations using model inversion techniques. Model stealing assaults, so implying Model stealing attacks try to copy how a target machine learning model works by asking it many questions and collecting the answers. With this input-output data, the enemy can train a dummy model to mimic the target model's actions. The attacker could use the original model's intellectual property without permission, revealing private parts of the model. One type of privacy attack is gradient inversion, in which an attacker rebuilds the original input data by using gradient information that was shared while a model was being trained. Usually taking place in federated learning systems, this approach aggregates gradients from individual devices to update a global model. Adversaries can recover the training data, resulting in privacy leakage, by iteratively adjusting a candidate input to match the observed gradients.

4.5 Membership Inference Attacks

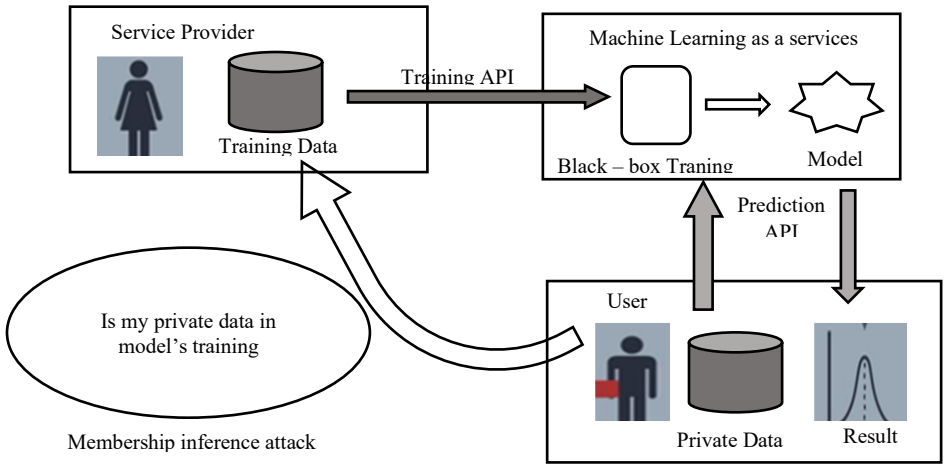

Fig. 6 Membership Inference Attacks

A membership inference attack is a privacy concern about the attacker's attempt to measure the specific individual data used to train a machine learning model. It's mainly concerned with Machine learning as a service, followed by uploading sensitive data for medical and financial records. This platform exposed the various details related to model outputs, followed by training sets against user privacy. The service providers should access the data handling, which means playing a critical role by enabling the analysis of the attacks. Followed by the proper safeguards, which contain differential privacy or output sanitizations. In the context of Large Language Models (LLMs), Membership Inference Attacks (MIAs) try to ascertain whether a particular data sample was included in the model's training dataset, which presents a privacy concern. Conventional MIAs have been researched for simpler models, but studies on LLMs show that the size and complexity of these models frequently make present approaches ineffective. Members guessing attacks (MIAs) try to fig 6 out if a specific record is in the training set for a particular model. Model privacy audits, training data memorization, copyright breaches, and test-set contamination can all be effectively investigated using MIAs. MIAs have been shown to have high attack performance, which may be because they remember a lot of training data. However, most studies are limited to classifiers or fine-tuning LM. How well current MIAs work on LLMs and their pre-training data hasn't been studied much. This study aims to investigate the obstacles associated with assessing membership inference assaults on LLMs, utilizing a variety of five frequently employed membership inference attacks. This paper presents MIMIR1, a central database for testing MIAs for LMs, including examples of different methods. This implies that existing MIAs' performance in prior contexts does not translate well to assaulting LLMs that have already been trained, presumably due to their inability to memorize member data. Additionally, we discover that the frequent overlap between members and non-members from natural language domains significantly impairs MIA

performance and leads to debate regarding the proper interpretation of membership.

4.5.1 Techniques and Methods

Membership inference attacks are used to determine the specific types of data points followed by the model training dataset; different types of techniques are used, such as shadow model training, binary classifier-based attacks, and metric-based attacks. The main aim of this shadow model is similar to the target model training data, which is also known as the membership status of each sample. The shadow model uses the processing stage of training attackers to predict the same data point as the target model.

4.5.2 Implications for Data Privacy

Data privacy is greatly affected by model inversion attacks, which target machine learning systems trained on sensitive or personally identifiable information. In such attacks, attackers use access to the outputs of a model especially confidence scores or class probabilities—to rebuild or deduce specifics about the underlying training data. Even in a black-box environment, this can cause sensitive information like user names, medical records, or biometric traits to be displayed. The issue is especially troubling for healthcare, banking, and tailored services models, where privacy infractions could lead to legal violations, identity theft, and loss of user trust. Moreover, inversion attacks question the idea that anonymized data is intrinsically safe as rebuilt inputs can usually be re-identified. Defining compliance with privacy rules such as GDPR and HIPAA and preserving the ethical and safe application of artificial intelligence technologies depend on so resisting these attacks.

V. DEFENDING AGAINST SECURITY THREATS

5.1 Robustness of LLMs Against Adversarial Attacks

By assessing the robustness of LLMs using two different approaches that concentrate on prompt perturbations, we hope to close this difference. The first approach uses character-level text attacks, as seen in Figure to randomly change characters in the input prompts to introduce character-level perturbations. Three popular sentiment classification datasets—Stanford NLP/IMDB, Yelp Reviews, and SST-2 are used to evaluate how resilient LLMs are to slight textual changes in prompts. A more advanced adversarial approach, jailbreak prompts, is employed in the second method. As seen in Fig. 2, jailbreak prompts are purposefully made to get against LLM security measures and cause them to provide answers beyond established use and ethical guidelines. We use a large dataset of 1405 jailbreak prompts gathered from various online venues, including as Reddit, Discord, and prompt-aggregation websites, to conduct this study. Different LLMs we tested have different levels of vulnerability to character-level and semantic-level adversarial attacks, as shown by our work. These results show that more excellent hostile training and better safety features are needed to make LLMs more resistant to these attacks. In summary, we have made the following contributions: The robustness of four well-known LLMs—GPT-4o, GPT-4, GPT-4-turbo, and GPT-3.5-turbo—is thoroughly assessed utilizing character-level perturbations and jailbreak prompts. We will examine the vulnerability of these models to adversarial assaults. Our results highlight the necessity of continued research and development to guarantee LLMs' secure and dependable implementation in crucial applications.

5.2 Detection and Mitigation of Poisoning Attacks

Effective mitigation against data poisoning attacks begins through robust data validation and filtering techniques. Ensuring the integrity of the training dataset is crucial. Various techniques are used for anomaly detection to identify suspicious data entries and compare the new data against establishment patterns. Various methods are used to detect the poisoning attack; SVM (Support vector Machine) is an ML technique used for classification and anomaly detection tasks. SVM is used as the binary classifier, which is the context of detecting the positioning attack to identify the cases that differed from the predicted data distribution. The key concept is to create a judgment boundary and maximize the margin between the class's spots through any types of fraudulent outliers injected by the poison data. SVM successfully identifies the potential toxic sample and then determines the integrity of the training data to find the model that should utilize the SVM capacity to understand the complex decision boundaries. SVM's ability to identify poisoning assaults depends on the kernel function and hyperparameters selected. Therefore, fine adjustment is required for optimal results. SVM's computational requirements can also be problematic for large-scale datasets, underscoring the necessity for effective optimization techniques. Researchers frequently combine SVM with other techniques, such as ensemble approaches or feature selection, to address these issues to improve detection accuracy and adjust to changing attack patterns. SVM remains an essential protective tactic in the ongoing efforts to prevent poisoning attempts and improve machine learning systems' security.

TABLE III COMPARISON OF VARIOUS POISONING TECHNIQUES

Learning Taxonomy	Advantages	Disadvantages
Conventional Traditional Supervised Learning	Algorithm as independent	Required extensive knowledge with unrealistic knowledge.
Conventional Unsupervised learning	Used for trustworthy situations	Vulnerable to outliers
Reinforcement learning	Attackers are required to undergo a limited amount of training. PCA approach was efficient.	Training Delays
Deep Learning	Utilized for Black box in realistic scenarios.	Inadequate performance evaluation.

Used decision trees to detect poisoning attacks. Decision trees create a tree-like structure that facilitates decision-making by recursively partitioning data based on unique

features. Decision trees can identify anomalies and outliers by examining distinct pathways within the tree structure in the context of poisoning attack detection. When making decisions, instances that follow odd or different paths could raise red flags and be classified as potentially poisonous data pieces. Since decision trees are interpretable, researchers can examine the characteristics and situations that lead to atypical decisions, providing valuable insight into possible poisoning attacks. However, decision trees are prone to overfitting even though they are interpretable and easy to use, especially when given noisy or unbalanced data. Pruning, regularization, and ensemble techniques (like Random Forests) enhance generalization and make decision tree-based detection more resilient to hostile attacks. Furthermore, finding important patterns and enhancing detection accuracy depends heavily on feature engineering and selection. Decision trees are helpful tools for spotting poisoning assaults and creating strong defenses that safeguard the reliability and integrity of machine-learning models across various applications.

Based on Table III interpret the Conventional unsupervised learning poisoning attack followed by feature selection principal component analysis. Another detecting method is deep learning. Conventional unsupervised learning poisoning attacks are considered cluster methods, which are used for Classical unsupervised learning and attempt to group data elements according to their similarity without necessitating any labeled information. In clustering, poisoning attacks can contort cluster development, causing data point misclassification and undermining the integrity of the clustering model. Assailants can inject malicious data to alter the cluster membership so that valid data points can be misclassified into incorrect clusters. Consequently, such applications as consumer segmentation, anomaly detection, and pattern identification may be affected by inaccurate insights and decisions. Feature selection method is employed. To perform unsupervised learning properly, you must identify the most critical characteristics that significantly affect how well the model performs. Poisoning attacks on feature selection involve introducing redundant or irrelevant features or modifying feature ranks to mislead the selection process. Attackers can deliberately insert noise or biased information into the system so that it selects features that are not optimal or are misleading. This reduces the model's efficiency, interpretability, and generalization, making it less applicable to tasks such as compression, dimensionality reduction, and data preparation. One popular technique for unsupervised feature extraction and dimensionality reduction is PCA (Principal Component Analysis). Transformation of the data into a new coordinate system highlights the important patterns. In PCA, poisoning attacks can involve injecting malicious data to change the magnitudes and directions of the principal components. Attackers can sabotage the reformed space by altering the covariance matrix of the data. This results in erroneous conclusions and jeopardizes the accuracy and credibility of the model in fields such as image recognition, signal processing, and outlier detection. Poisoning attacks in Deep Learning In deep learning, poisoning attacks involve intentionally introducing corrupting material into the training set. These malicious samples are carefully made to change how the model learns, leading to skewed or wrong predictions during inference. Poisoning attacks are tough to spot because they target the learning processes instead of software bugs like most attacks. Diversified strategies are needed to lessen the risk of deep learning poisoning assaults. Various methods, including adversarial training, data cleaning, and model stability testing, make the model more resistant to bad situations. Furthermore, increasing awareness of potential weaknesses and integrating security considerations during model development and deployment are essential to protecting deep learning systems. It is impossible to overestimate identifying and stopping poisoning assaults because deep learning is at the forefront of AI research and applications (Srinivasan et al., 2024).

5.3 Defense Mechanisms for Evasion Attacks

Several defense strategies have been suggested; machine learning (ML) approaches seem promising. Data-driven IDSs can be classified as supervised learning, semi-supervised, unsupervised, self-supervised, and reinforcement learning. Designed to find evasion attacks, model-based defense mechanisms can use physics to increase data efficiency. Still, poisoning attacks, a growing concern, have much less effect on the efficacy of ML-based IDS. Unlike an evasion attack carried out during deployment (test time), a poisoning attack might mislead the model training by changing the training data, producing a fake deployment model. Defining against poisoning threats is difficult and understudied for power system cybersecurity. Moreover, the situation of combined poisoning and evasion attacks has not been covered in very few studies, hence the body of knowledge is very lacking. The difficulty is to reason on the spread of error caused by poisoning attacks to assess time performance and thereafter build a strong learning framework for such error propagation. The present work starts the first investigation in this crucial line. We use bilevel optimization as our method. The latest line of study on end-to-end optimization inspires the approach design. Its eventual use during test time offers a moral approach to building a model's training. Supervised, semi-supervised, unsupervised, self-supervised, and reinforcement learning are several ways IDSs can be classified. Designed to find evasion attacks, model-based defense systems can use physics to increase data efficiency. Still, poisoning attacks are a growing concern they have much less effect on the efficacy of ML-based IDS. Unlike an evasion attack carried out during deployment (test time), a poisoning attack might mislead the model training by changing the training data, producing a fake deployment model. Defining against poisoning threats is difficult and understudied for power system cybersecurity. Moreover, the combined poisoning and evasion attacks situation has not been covered in a lot of research; hence, the body of knowledge is very lacking. The difficulty is to reason on the spread of error caused by poisoning attacks to assess time performance and thereafter build a strong learning framework for such error propagation. The present work

starts the first investigation in this crucial line. Our approach is based on bilevel optimization. Inspired by the latest line of study on end-to-end optimization, which offers a principled approach to building a model's training given their consequent usage during test time, the methodological design

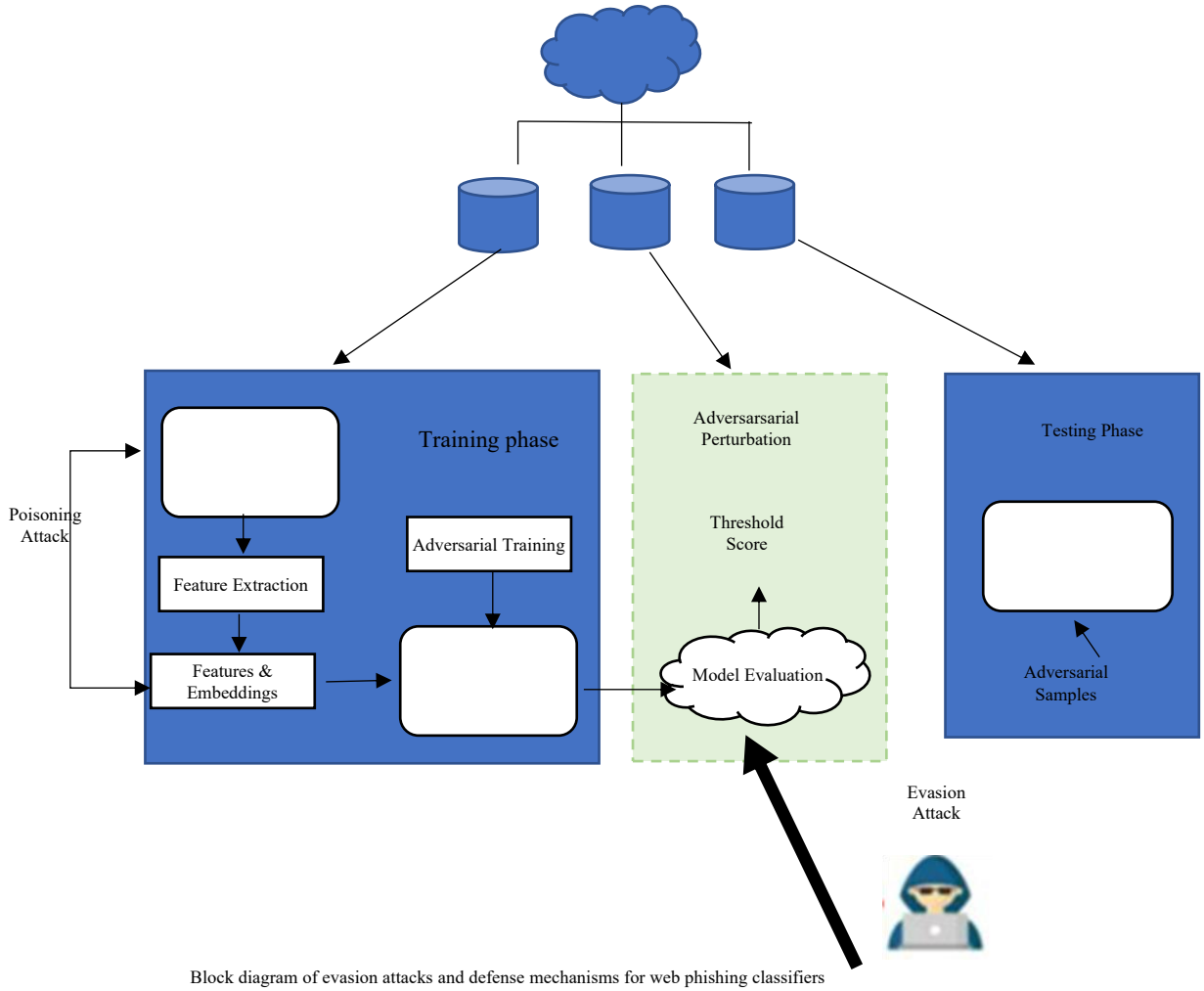

Fig. 7 Block diagram of Evasion attacks and defense mechanism

The above fig. 7 describes a block diagram of an Evasion attack in a defense mechanism for a web phishing classifier. It also explains such systems, which are followed by training and testing phases. Data analysis feature extraction is embedded through poisoning attacks during the training phase. Adversarial training is employed through defense mechanisms to improve robustness. To evaluate the stage, we introduced the antagonistic approach to assess the performance under potential attacks. During the test phase, evasion attacks are launched and used by the various samples related to bypass detection and challenged through classifier accuracy. The overall system is linked to central cloud data and demonstrates the adversarial strategies in defense impact phishing detection efficacy.

5.4 Techniques to Mitigate Model Inversion Risks

Several ways to protect against model inversion threats occur when malicious actors try to reassemble private training data using the model's outputs. Differential privacy, which involves introducing noise into the training process or outputs to obfuscate specific data points, is a highly effective method. Changing the output, limiting the accuracy or detail of model answers, is another way to lower exposure. Restricting suspicious or excessive queries is a common tactic in inversion efforts; to counter this, utilize authentication, rate limitation, and access restrictions. Furthermore, regularization techniques like weight decay and dropout can reduce the training data that needs to be learned by preventing overfitting. By putting knowledge into a smaller, less thorough model, model distillation can make it even harder to remember things. Combining these methods with anomaly detection and ongoing observation guarantees strong protection against model inversion and protects private data in machine learning systems.

5.5 Privacy Preserving Machine Learning

Many privacy-enhancing methods centers on letting different input sources train ML models together without exposing any of their private data. Cryptographic methods and differentially private data release (perturbation techniques)

were the tools of choice to accomplish this. When protecting against membership inference attacks, differential privacy shines. In conclusion, restricting the model's prediction output (e.g., to class labels only) can reduce the efficiency of model inversion and inference about membership attacks, as mentioned above. Methods Based on Cryptography Applying cryptographic techniques to train and test ML on encrypted data becomes possible when a particular ML application needs data from numerous input partners. Data owners contributing encrypted data to calculation servers reduced the challenge to a secure two- or three-party computing setup, which improved efficiency in several of these methods. Not only are these methods more efficient, but they also don't necessitate that the people providing feedback stay online. Horizontally partitioned data is the focus of the majority of these methods: Every data owner has compiled the identical collection of attributes for various data objects. Case in point: facial recognition, where users can input several feature vectors derived from their images to train an ML model specific to their face. In every one of these scenarios, every data owner is pulling out the identical set of features. The most common cryptographic methods for PPML realization are homomorphic encryption, jumbled circuits, secret sharing, and safe processors. Encryption using Homomorphism: Computation of encrypted data is made possible by fully homomorphic encryption. Simple arithmetic operations like addition and multiplication can be utilized as building blocks for more complicated arbitrary functions. It is expensive to bootstrap the cipher text often, which means refreshing it because of noise that has built up. This is why additive homomorphic encryption methods were mostly used in PPML approaches. Encryption systems like these restrict operations to addition and multiplication by plaintext. In order to make additive homomorphic encryption more versatile, protocols were created to make it possible to compare two encrypted values or execute secure multiplication and decryption operations. One common method is to "blind" the cipher text, which involves adding an encrypted random value to the encrypted value that needs to be protected. To make additive homomorphic encryption, which relies on all of the aforementioned methods, more efficient, data packing techniques were devised to allow the encryption of multiple plaintext values using a single cipher text. The data owners in this system communicate their encrypted data to the service provider (SP) using the public key of a privacy service provider (PSP). Customers (the data owners) can take advantage of the PSP's compute and privacy services, while the SP's storage and calculation capabilities allow it to create private suggestions. Coordination between the SP and PSP is essential to the safety of such a system. The non-collusion assumption is reasonable because the SP and the PSP could be distinct companies, which makes sense given the services they provide. While SP and PSP are involved in computation in this system, data owners are involved in both input and results. Circuits in Disarray: By transforming the computation into a garbled circuit and transmitting it with her own garbled input, Alice can obtain the outcome of a function computed on her own inputs and Bob can do the same in a two-party setup. Without Alice discovering Bob's secret input (e.g., via oblivious transfer), Bob receives the jumbled version of his feedback. Following this, the CSP generates a jumbled circuit and transmits it to the evaluator, who in turn receives the jumbled form of the intermediate shares. The evaluator can construct the necessary ML model(s) using the jumbled circuit and its jumbled input. As opposed to the training and testing stages, the testing phase is the primary focus of several PPML methods. The goal was to make it possible to test additional samples without jeopardizing the provided samples or the ML model. A technique for keeping a secret under wraps is known as "secret sharing," it involves assigning a "share" of the secret to several people or organizations. Shares are meaningless in isolation, but the secret can be revealed when pooled together. Each user exchanges their Diffie-Hellman private key and user-specific secret with others to compensate for users who drop out before the protocol is finished.

VI. Real-world Case Studies of Security Attacks on LLMs

6.1 Adversarial Attacks in NLP Systems

Adversarial attacks in NLP systems should demonstrate the effectiveness of a large language model in generating valid and natural adversarial examples through the real-world level of substitutions. This approach mainly fig 9 focuses on world-level modification to encompass various adversarial attacks, including adversarial patches, universal perturbations, targeted attacks, and transferable attacks. Adversarial patches mean LLM should generate contextual and relevant text snippets, meaning input causes misclassifications. Universal Perturbations, which means it utilizes the LLM to create the text perturbations for effectiveness, followed by the multiple input, which is target multiple to the target model. Targeted attack means employing the adversarial example as a specific misclassification leveraged through a deep understanding of language and context. Transferable attacks, which means exploiting the LLM broad knowledge through generating adversarial examples, are effective across the different model architecture domains. A novel approach helps generate the adversarial NLP patches for using LLM, as described in the following fig 8. This approach represents fundamental concepts with analysis to conceptualize and create adversarial examples of data. Additionally, traditional models are related to simple word replacement followed by various approaches. It allows for generating the adversarial examples integrated with surrounding the text and also thoroughly detects the various challenges (Esmradi et al., 2023).

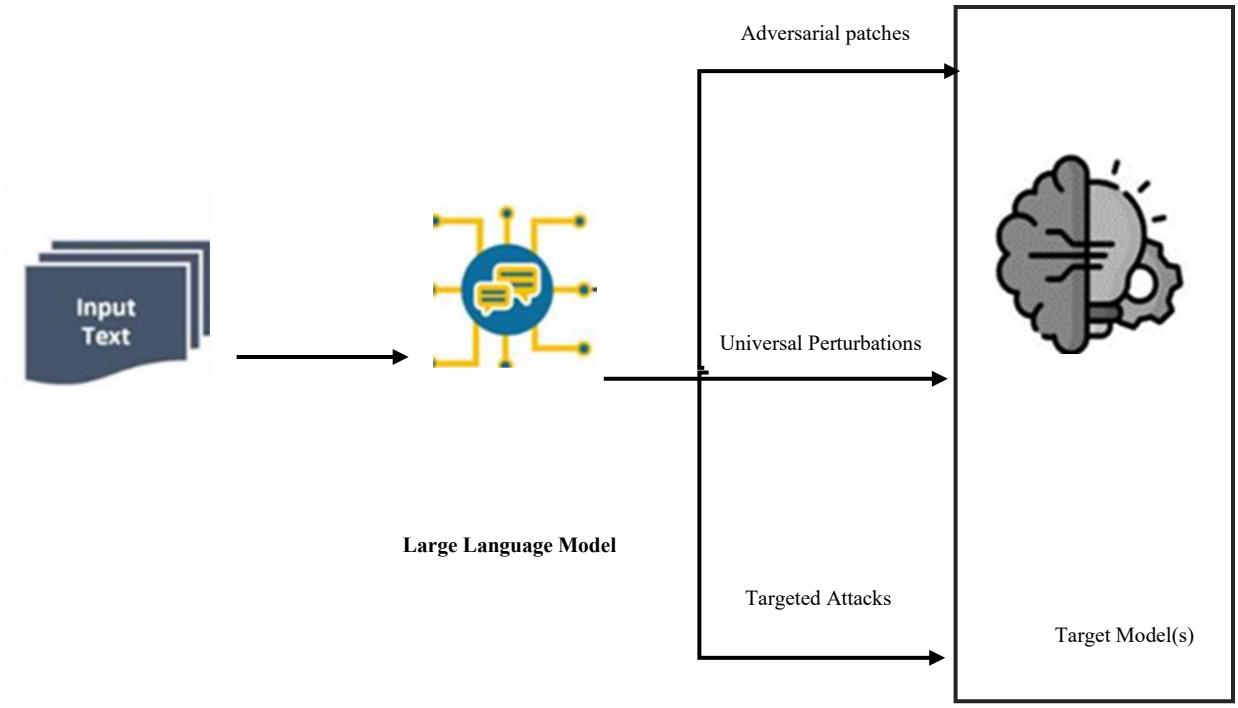

Fig. 8 Adversarial Attack for NLP System

6.2 Poisoning Attacks on LLMs in Production

This section describes real-world examples of data poisoning attacks. These attacks not only compromise the integrity of the AI system but also analyze the significant values through financial reputation and operational risks to the business. Here, an analysis of the mechanism for safeguarding enterprises in LLM applications is also conducted. Here, it also describes the several real-world impacts related to examining the specific instances of data poisoning affected by the various companies. Instant Injection An immediate injection attack affected a GPT-3-based Twitter bot run by a recruiting business named Remoteli.io, leading to a serious incident. The clever introduction of malicious inputs into the bot's functioning caused it to divulge its original prompt and respond inappropriately to debates about "remote work." Along with seriously compromising the startup's social media communication capability, this breach presented legal and reputation hazards. The episode draws attention to the serious flaws in artificial intelligence systems and the far-reaching consequences such attacks might produce for public confidence and corporate operations. Code Injection Maliciousness Attackers got into AI-based code makers by adding destructive code during the training data phase, which is a worrying security breach. Attackers posted 100 poisoned models to the Hugging Face AI platform, where the infected models might insert harmful code into user systems, facilitating this manipulation. The event draws attention to a serious supply chain risk since these poisoned models can be unintentionally included in other systems and uses. The hack has shown how urgently improved security policies and careful screening of artificial intelligence models are needed to stop such events in the future and, therefore, shield consumers from significant security threats. Manipulations of Output Using ChatGPT, New York attorney Steven Schwartz clearly showed output manipulation throughout the legal procedures of Mata v. Avianca. Schwartz used AI to study the law, hoping it would give him accurate results. Regrettably, the AI fabricated false court citations and case law and employed them in its responses. This error caused by the processing and generation mechanisms of the AI can be determined by its training data, which is an instance of output manipulation where the output of the AI was misleadingly altered and produced incorrect legal documents. Schwartz's use of these defective AI outputs without sufficient vetting seriously impacted his career and raised important questions regarding the reliability of AI in legal research. A more general debate on the ethical implications of the use of AI tools in sensitive and significant decision-making areas was also prompted by this abuse, in addition to a court case.

Unsensitized Scraped HTML: The Case To highlight the issue, a company instructs a chatbot to serve customers by learning enormous amounts of text data by web scraping. If the collected HTML content from the internet isn't cleaned sufficiently, it might contain quietly taking place destructive scripts or deceptive metadata. For example, a seemingly harmless blog post might contain encoded HTML comments or program tags the LLM interprets as legitimate data. Malicious prompts inserted into the LLM's data for training might be the result.

For example, an unsensitized script tag on a scraped webpage can contain a payload that gently influences the chatbot's answers to favor a particular good or service or, worse, provide answers, including phishing links. These injections might not immediately present concerns during the model training stage since they can replicate typical fluctuations in data, so they are included in the operational framework of the

LLM. The result is a model that has been hacked and carefully taught to do things that fit the attacker's plan. Over time, this manipulation can continue without being noticed, slowly changing business choices and customer interactions until big problems require a closer look.

It also explains safeguarding against data poisoning and provides the best practices in AI security. To mitigate the threat of data poisoning, which means protecting and implementing AI-specific measures. Straight for data validation protocols means ensuring that each training dataset is scrutinized for integrity and authenticity. To check the data for accuracy and relevance to analysis for manipulations. Regular audits and updates identify the subtle discrepancies that should indicate the tampering. Update means ensuring that the AI modes are incorporated with the latest security features and that new threats are adopted. A layered security strategy contains software solutions and operational protocols. A software solution means implementing the software to enhance security. The operational protocol should include encrypting data sources to enforce access control and segregate the training data through operational data to prevent unauthorized access and contaminations. Employee education means training the employee in AI security and data poisoning to analyze the defenses. When implementing your LLM depending applications in your firm, the thread of data poisoning becomes a significant concern. We need strong security measures and close monitoring to protect against the dangers of bad people slowly changing the data that our Large Language Models (LLMs) use. Some of these are strict validation of data rules, typical simulation audits, and advanced preventative measures like Red Collaboration. Giskard specializes in providing thorough testing systems for different artificial intelligence models, including LLMs.

6.3 Vulnerabilities in Chatbots and Virtual Assistants

Large language models like GPT power chatbots and virtual assistants that revolutionize human-computer interaction but come with serious limitations. Prompt injection attacks in which malicious users control the model's input to bypass safety filters or trigger undesirable behavior are causing much concern. An attacker could, for instance, add hidden instructions within a prompt to allow the model to bypass previous rules or constraints. Most closely related to this is jailbreaking, where one designs inputs so that the model performs restricted actions, typically by employing encoded, inverted, or cleverly written commands. Data leaking is another crucial flaw where the model might inadvertently remember and reveal personal information from previous interactions or training data. This raises serious privacy issues, especially when the virtual assistant is integrated with systems handling personal or confidential data. Adversarial inputs of examples designed to mislead or trick the model can also result in wrong or damaging outputs. In sensitive domains such as finance or medicine, these can particularly be dangerous. Additionally, pressing issues involve disinformation and impersonation. While fluent and persuasive, LLMs are not inherently capable of distinguishing right from wrong information. Therefore, they can be manipulated to produce spurious identities or post misleading material, which could result in false information being released or social engineering becoming more accessible. Users tend to accept the model's outcome at face value, even though it may be inaccurate or hypothetical, because too much reliance on LLMs makes the situation worse. Insecure API connections also put systems at risk, especially when chatbots are connected to backend operations such as user databases or payment gateways. Attackers can utilize these connections to carry out malicious orders or gain illegal access. Another problem is the lack of explainability of LLM behavior. Since such models are black-box, it's difficult to determine why they gave a particular response; this makes debugging and security improvements even more of an uphill task. Developers and companies must employ vigorous safety protocols to mitigate these risks. These include implementing input and output filters, periodic red-teaming exercises, human-in-the-loop training methodologies, reinforcement learning with human judgments (RLHF), and educating users on the limits of artificial intelligence systems. In addition, transparency in AI action and restricted access to sensitive features guarantee that virtual assistants and chatbots are trustworthy and secure.

6.4 Attacks in Large-scale Deployment of LLMs

Large Language Models (LLMs) are perfect for tasks ranging from creating human-like text to comprehending real language. Still, enormous power carries enormous responsibility. Protecting private data and stopping misuse is essential when using LLMs. In LLM deployment, safety refers to ensuring the models perform as planned and do not generate unanticipated or adverse effects (Sakib et al., 2024). This cover ensuring the models are used sensibly and suppressing prejudiced, misleading, or damaging material. Conversely, security ensures the models and data are controlled against illegal access, breaches, and hostile behavior. Part of protecting an LLM model is ensuring that the infrastructure is safe, that encryption is used, and that only authorized people can see private data. Secure and safe placement of LLMs is crucial for various reasons. First let us name those. Unprotected LLMs could be utilized improperly, such as phishing attempts or fabrication of fraudulent information. LLMs routinely generate information from vast amounts of data, sometimes including sensitive personal information. Therefore, avoiding unauthorized access to this data is very important for privacy and security (Ando et al., 2012). Users' confidence depends on following laws including GDPR or CCPA to avoid legal repercussions or eroding confidence Keeping the main causes in mind, let's now try to grasp why safe and secure deployment is threatened by present LLMs. The fast growth of advanced LLMs comes with a lot of risks and challenges that are very different from those faced by standard machine learning models, especially when it comes to moral issues and how they can be interpreted. Let us examine these two features individually.

Here also explained the various types of key challenges related to deploying LLMs as The numerous challenges that

come with deploying LLMs must be thoroughly investigated if their efficient and safe usage is to be guaranteed. The most important thing is to keep the data safe. A lot of data, some of it private, is put into these big files while they are being trained. Keeping this data secure throughout training and inference is critical to preventing unauthorized access and security breaches. Modern methods include data encryption, safe data storage, and rigorous access control policies, which are what we must use to safeguard this material. Data encryption converts data into a coded form only accessible to those possessing the necessary decryption keys. This guarantees that, even in the case of interception, the material is difficult for illegal users to interpret (Sumartono et al., 2024). By protecting both the electronic and actual places where data is kept, safe data storage keeps it from getting lost, changed, or accessed without permission. Assume one storage device will fail with a certain probability. If that happens, you can use backup tools like RAID (Redundant Array of Independent Disks) to ensure you don't lose any info. Clear Access Control Policies: By regulating who may see and use what in a system, access control processes guarantee that confidential information is accessible only to those who should have it. One straightforward approach to protect LLMs is using role-based access restrictions. Still, another big obstacle is model bias. In open-source models, if you have not trained yourself, biases from the training set of LLMs may be inherited and generate unjust or discriminating results (Hassanin & Moustafa, 2024). This is a big issue, especially in highly regulated sectors like banking and healthcare, where skewed results could have pragmatic consequences. We must thus be cautious when choosing our datasets, implementing bias detection strategies, and applying mitigating approaches to ensure the models produce fair and equitable outcomes (Bathala & Babu, 2024). The rules help you to handle bias in Large Language Models (LLMs). Pre-processing, processing, and post-processing stages make up Incorporate Bias Detection Methods. You should consider data auditioning and label distribution analysis during the pre-processing stage. Processing stage: think about adversarial debiasing to prevent the LLM learning from biased connections. Outcome testing and counterfactual fairness testing are two methods of assessing if predictions vary.

In this section, the best practices for secure deployment are detailed. When possible, anonymize data to remove identifying information and minimize the chances of exposing sensitive data. Explore the use of differential privacy techniques to introduce noise-to-query outputs that protect individual data points from being traced to individual users. Regularly test and verify your models to ensure they create equitable outcomes and function as intended. Integrate performance testing within your development cycle to ensure your model consistently meets the desired results. Never underestimate security testing; identify potential vulnerabilities before they turn into issues. Develop continuous notifications and logging to monitor for anomalies or security issues. It will assist you in staying compliant and responding quickly to issues. Implement an act of tracing to ensure your deployment is stable and secure in the coming years. Implement role-based access control (RBAC) to ensure users can only view the data and features necessary for their roles. Strengthen security even more and guard against illegal access by including multi-factor authentication (MFA.). Plan regular audits and improvements, among other things. Penetration testing, access record analysis, and security audits are all part of this. Maintaining thorough audit records will help you monitor all modifications and access attempts, offering an unambiguous system operations record. Following these best standards will help you to guarantee the effective and safe operation of LLMs, thereby protecting private information and maintaining user confidence (Kumar et al., 2024).

VII. Ethical and Legal Implications

7.1 Ethical Challenges in LLM Security

Improving Our Knowledge of LLM Security and Privacy Concerns In order to provide results, forecast next-word sequences, or categorize data, LLMs process massive volumes of data. While doing so, they frequently come into contact with sensitive information, proprietary data, and personal details (Gan et al., 2024). Nevertheless, substantial dangers are also posed by this interplay. Data leakage, adversarial attacks, bias propagation, deep fakes, and disinformation are some of the primary topics covered in this article. Data Leakage: Whether it's accidental inference during training or malicious attacks, LLMs trained on large datasets can accidentally reveal private data from the training data. Malicious Attempts (Rathod et al., 2025). The safety and soundness of the model can be compromised if malicious actors use LLMs to generate biased or destructive content. The use of LLMs trained on biased datasets increases the likelihood that discriminating decisions or outputs will be produced, as well as the likelihood that preexisting social prejudices will be amplified (Chen & Madisetti, 2024). Deep fakes and Misinformation: Using LLMs to make fake material that looks real is a big problem when we're trying to stop misinformation and people losing trust in the media. An incident in LLM Privacy and Security includes data leakage during rollout, malicious use of LLM outputs, and the spread of bias in the language model. Data Exposure in LLM Rollouts Data leakage, the accidental disclosure of sensitive information during contact, is a prevalent privacy risk with LLMs (He et al., 2024). For instance, some LLMs have been known to accidentally share private information from their training samples when users ask them to. If models are trained on real-life conversations or sensitive datasets, this shows how important it is to have tighter rules for handling data. Subversive Control of LLM Results It is also possible to conduct adversarial attacks on LLMs by manipulating the inputs so that the model generates false or misleading results. Subtly changing input prompts is one way these assaults might cause the model to produce biased, incorrect, or potentially harmful outputs. For instance, there are issues about the integrity and security of content produced by LLMs because hostile parties might provide adversarial inputs to an

LLM, which produces damaging language or lies (Das et al., 2025). The Prejudice Spreads in Syntactic Models The prejudice inherent in the data used to train LLMs is often present in the outcomes. Models like GPT-3 have been involved in several high-profile incidents for the potential to unknowingly employ discriminatory phrases or stereotypes while explaining gender, religion, or ethnicity. Discriminatory outputs of AI-based decision-making systems might reinforce inequality or bias in recruiting, law enforcement, and finance fields, along with the issues these biases raise in normal interactions. Strengthened safeguards, rigorous data curation, and sustained improvement in model transparency and explainability are called for with urgent need to address the security, fairness, and privacy issues posed by LLM (Kumar et al., 2024).

Best Practices for Secure Deployment Whenever you can, anonymize data to eliminate personally identifiable information, lowering the danger of revealing private information. Use differential privacy methods to provide noise to query outputs, safeguarding specific data points from being connected back to particular users (Rathod et al., 2025). Test and validate your models often to be sure they generate fair results and operate as expected. Including performance testing in your development process can help ensure that your model regularly satisfies the intended outcomes. Never undervalue security testing; find possible weaknesses before they become problems. Create ongoing monitoring and logging to track any anomalies or security concerns. This will help you to keep compliant and react fast to problems (Liu et al., 2025). Create a practice of tracking to guarantee that your deployment is stable and safe over the years. Use role-based access control (RBAC) to guarantee users may only access the data and features required for their responsibilities. Strengthen security even more and guard against illegal access by including multi-factor authentication (MFA.). Plan regular audits and improvements, among other things (Wang et al., 2019) Penetration testing, access record analysis, and security audits are all part of this. Maintaining thorough audit records will help you monitor all modifications and access attempts, offering an unambiguous system operations record. Following these best standards will help guarantee LLMs' effective and safe operation, protecting user confidence and private information (Andraško et al., 2021).

7.2 Legal and Regulatory Aspects of AI Security

Large language models (LLMs) show skills that might significantly affect legal activities. Their mastery is demonstrated by events including a model like GPT-4 passing the American Uniform Bar Examination and the All-India Bar Examination, where it exceeded past versions of LLMs and topped the average scores of human test-takers (O'Sullivan et al., 2019; Katulić, 2020). This achievement shows how good the model is at comprehending and thinking about the law. It also hints at a future where LLMs can do repetitive analytical jobs automatically, freeing up lawyers to focus on strategy and more difficult legal work. This milestone shows how better the model is over past LLMs and individuals and suggests its possible influence in transforming legal practice (O'Sullivan et al., 2019).

Studies on the market for commercial artificial intelligence applications estimate which AI could perform up to 44% of administrative chores (Kemp, 2018). This claim arises from the idea that a good fraction of legal labor consists in the study of complex yet orderly documents—a choreography fit for the algorithmic accuracy of LLMs. If these predictions materialize, LLMs might become quite significant in contract analysis and due diligence, thus liberating lawyers to focus on more complicated aspects of their job. Leverage of artificial intelligence would enable companies to re-imagine their economic models, migrating from a quantitative to a value-based service model, thus building a legal framework that provides customer satisfaction along with results with an over billable time first priority (Sumartono et al., 2024). Legal services expert and futurist, Richard Susskind, believes law firms with artificial intelligence at the helm are shifting from time-based to value-based billing. The transition aims to enhance cost-effectiveness to benefit legal practice and consumers alike. LLMs might transform various areas of legal practice. Four main aspects of the legal profession can significantly benefit from AI: research, document production, dissemination of information, and sophisticated analysis. This assistance extends beyond mere work automation to also attempt to enhance the quality of legal work (Subramanian, 2017). For instance, LLMs have the ability to make case retrieval tools more precise by modifying and enhancing search queries. Such new technologies accelerate the retrieval process and allow lawyers to access the most recent and relevant legal precedents and documents, which is vital in preparing cases and representing clients appropriately (Neupane et al., 2024).

Envision a sophisticated Online Dispute Resolution (ODR) system that processes case information using an LLM. To speed up resolutions in high-volume, low-stakes disputes, LLMediator, for example, leverages GPT-4's processing power. To facilitate negotiations to proceed more smoothly, it simplifies user interactions, prepares responses automatically, and enters into discussions on its own (Kavibharathi et al., 2021; Dhruvitkumar, 2024). This platform demonstrates not only the potential to enhance the ODM process but also to expand access to justice significantly. Encouraging initial estimates suggest that LLMediator is paving the way for future innovations of AI-enabled legal mediation, thus highlighting the revolutionizing potential of artificial intelligence on the legal sector. With regards to arbitration, LLM can redefine the steps of disclosure and document creation as it can set up and scrutinize vast sets of data, revolutionizing them. AI is likely to offer efficiencies that substantially reduce the cost and time generally associated with such stages by digging through documents and identifying relevant items. LLM software can also accelerate the examination of pleadings and submissions by creating brief abstracts and providing initial counterarguments based on the available evidence (Biasin et al., 2024). This assists attorneys in quickly grasping intricate

submissions. While artificial intelligence may be quite effective for certain work, a proper examination by attorneys is still completely necessary to avoid presenting the tribunal with false evidence.

With the courts, the contract interpretation ability of LLMs is a significant shift at the judicial end. LLMs are able to support lawyers and judges to understand the hidden meanings and drives behind words through language and context analysis of a contract. The technology, as helpful as it is in enhancing human judgment with a richer insight into the terms of the contract, does not eliminate its requirement. This can lead to decisions that honor the actual agreement between parties, ensuring justice and compliance with the spirit and letter of the contract. Beyond mere contractual issues, LLMs may influence more general court rulings. Judges currently review each case on its merits, but LLMs may offer the necessary guidance to ensure a more uniform interpretation of laws and precedents (Judge et al., 2025). This auxiliary tool supplies uniform legal meanings while allowing human touch in final decisions, and not an adjustable-fits-all justice system. Such a balance can perhaps make the judicial process more efficient and equitable, and reflect a move towards a fairer, newer, and better legal system (Jing et al., 2021). LLMs do not come without challenges though. One should be cautious relying too heavily on LLMs. There are dangers, such as when a New York judge scolded attorney Steven A. Schwartz for drawing on points from ChatGPT in a case involving the statute of limitations. This occurs when LLMs invent information that appears genuine but is not. The complex, jargon-laden writing style of legal texts, the large number and non-digital nature of many documents, and the extensive privacy issues related to handling sensitive data all contribute to exacerbating difficulties. If this occurs, closed AI systems can assist by safeguarding client privacy and user data (Anthis et al., 2024; Chen et al., 2024).

7.3 Accountability in LLM Vulnerabilities

Large language models (LLMs) have significantly extended natural language processing to include Google's Gemini, Meta's LLaMA, and OpenAI's GPT-4, hence making a wide range of application areas possible across education, medicine, customer care, and much more. All these models come with a range of shortcomings which raise significant ethical and legal concerns despite their potential benefits. The question of who bears responsibility when these technologies fail or cause harm becomes more serious as their use expands more widely. Technical limitations to end-user misuse set the parameters for the varieties of LLM vulnerabilities. The repetition and exaggeration of societal biases embedded in the training data are among the most pressing problems (Barman et al., 2024). Even if they don't mean to, biased input data might cause these models to produce discriminating or objectionable content. Another critical concern is hallucination, which occurs when the model self-assuredly produces inaccurate or deceptive data. This can be especially risky in fields with a lot at stake, such as healthcare or legal advice. LLMs could also be vulnerable to cybersecurity threats, including vulnerability to prompt injection attacks and capability for generating phishing content or fake information (Wei et al., 2024). Having been deployed, the autonomous nature of LLMs complicates keeping track or regulating all their output, thereby complicating attempts at responsibility. From an ethical point of view, developers and deployers of LLMs are under a mandate to anticipate and minimize adverse effects. AI needs to be explainable and accessible; users need to know when they are interacting with AI and be able to infer, to the best of their knowledge, how the model constructs its responses. Denying this knowledge undermines user informed permission and control. In addition, the use of LLMs without safeguards can compound digital inequalities disproportionately harming disadvantaged or underrepresented populations who may lack the resources to engage with AI systems or to actively defend themselves against abuse. Additional complexities exist regarding legal accountability.

Certain accountability frameworks have adapted to address these challenges. Model cards, datasheets, and audit trails can be employed to monitor the interactions and inferences of the AI, as well as their history and limitations. The practice of provenance monitoring, tracking where data originates and how it was trained, is also increasing. The use of human-in-the-loop technologies, in which human oversight is sustained through the deployment of AI, assists in ensuring safety and ethicality in sensitive situations. Real-world examples further prove this. For instance, the tension between security and openness is illustrated by OpenAI's GPT models. Concerns regarding misuse of powerful language tools, including Meta's LLaMA models, to develop toxic content made people discuss whether it is right or wrong to make them open source (Sarker, 2024). At the end, serious moral and legal concerns arise from LLM vulnerabilities, making it necessary for a transparent structure of accountability. Without those guidelines, the probability of intentional and unintentional damage is still higher. Technologists, ethicists, legal experts, regulators, and civil society representatives will have to collaborate with each other in the future in order to ensure that AI is being developed responsibly. The only method of ensuring LLMs are being used safely, ethically, and value-driven is by such cooperation (Andraško et al., 2021).

7.4 Ensuring Transparency and Fairness in LLM Systems

Artificial Intelligence (AI) has revolutionized various sectors, such as finance and healthcare. But with the growing influence comes a major hurdle (Dai et al., 2024) artificial intelligence bias. "Artificial intelligence bias" refers to AI systems' widespread and unfair prejudice, usually because of biased data or algorithms. This bias could lead to skewed results that poorly affect business decisions and society's perceptions (Freiberger & Buchmann, 2024). Keeping fairness and trust in artificial intelligence systems thus hinges on identifying and mitigating bias. A key strategy for this issue is LLM monitoring, or testing big language models for biases to make AI applications fairer. Bias measurement in artificial intelligence systems relies heavily on fairness metrics. Demographic parity, which ensures fair treatment

for every demographic, and equalized odd, which measures the validity of AI predictions for different demographics, are two common metrics used. Estimating bias and informing mitigating steps rely on these metrics (Kumar et al., 2023). A comprehensive toolkit intended to detect and mitigate bias in AI models, AI Fairness 360 (AIF360) Developed by IBM, AIF360 offers a set of measures and algorithms to measure fairness. Developers can apply it to recognize biases at various stages of AI model creation and obtain actionable recommendations for ongoing enhancement. This also shows that Multiple organizations have successfully used fairness measures and tools to reduce biases in their AI frameworks. For instance, included demographic parity in hiring algorithms, resulting in a more equitable hiring procedure (Sun et al., 2021).

Another example was when a banking company ensured that its loan application procedure treated candidates evenly, without prejudice to history, using AIF360. Adversarial debiasing and rewetting use debiasing techniques. Rewording is one technique used in balancing representation between multiple groups by modifying the value of different data points. It results in a fairer dataset by allocating weights to underrepresented groups and reducing bias within AI model outputs (Singh, 2025). Training AI models to reduce bias through adversarial methods is referred to as adversarial debiasing. This process employs a secondary model to nudge the primary AI model to be more equitable without compromising accuracy. Prepare data to achieve balanced representation across groups. Reweighting or adversarial training can be applied to minimize bias. Apply fairness metrics to evaluate the compromised model and ensure improved equity. Operational observations and best practices for data lineage, algorithmic modification, and pretreatment of data were also witnessed here. Data preprocessing treatment determines if it causes or reduces bias (Haurogné et al., 2024).

Among the best practices are guaranteeing diverse data gathering, normalizing data to prevent biased distributions, and using techniques such as oversampling or under sampling to balance datasets. Algorithmic adjustments, therefore, suggesting that another effective method for reducing prejudice is altering algorithms to incorporate fairness constraints. Tracing the causes of bias involves tracing the history of data. Recording the path of data from capture to release makes it easier for organizations to identify possible biases and intervene. In this case, it also outlined the various challenges and solutions, for instance, A dearth of diverse datasets, poor technical expertise, and complexity in identifying minor biases, among the issue's bias detection and prevention face. These challenges can hinder the attempts to create fair artificial intelligence systems (Ayyamperumal & Ge, 2024). Organizations employed diversified development teams to introduce diverse perspectives, and multiple approaches were seen to manage these challenges. Sponsoring training programs will enhance technical skills and decrease bias. Utilize regulations and debiasing tools to speed up the process.

VIII. Future Directions in LLM Security

8.1 Research and Development of Secure LLM Architecture

The below fig 9 explains the architecture of Tstream LLM followed by LLM with a secure and adaptable stream processing system. Here also defined the various components involved through real-time data streams to continue learning and adaptation in LLMs. This architecture is divided into various components, such as traditional types of time series stream processing, which helps to develop the LLM integration. At the beginning stage with pipeline concepts, Data ingestion involves acquiring continuous data through various sources. These streams are passed through the stream processing module, which is performed by real-time processing transmitted through investigated data. These modules should handle the user interface requests and responses and be indicated as the primary interface among the data and user interactions. The processing data stream is noted as a real-time adaptation and learning module; it's the key innovation of the TStream of LLM. It is able to enable the system to adopt the LLM dynamic responces to the patterns. It also ensures that the model is contextually aware of the recent information. To introduce the transaction requests and responces to ensure the correctness of the model updates with the inference process. The component should access and interact with LLM to maintain the model parameters and metadata. The state of these management components should act as the persistent and metadata.

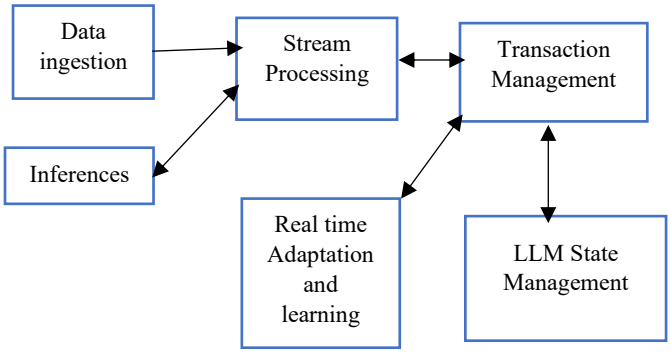

Fig. 9 Security LLM Architecture

8.2 Emerging Threats and Evolving Attack Methods

Here, we explained the various threats used in attack methods, such as sophisticated malware and ransomware. Ransomware, which means these types of attacks are increasing and prevalent. These types of attackers are in demand of highly used techniques like data exfiltration and threats through public disclosure (Motlagh et al., 2024). Being regularly involved with malware containing new types and variants makes it challenging to detect and prevent. Phishing and social engineering mean phishing attacks are analyzed as significant threats, and attackers are used to increase techniques to reveal sensitive information. Social engineering attacks should exploit human psychology to manipulate the individual to have confident information through performed by compromised security. Supply chain

compromises, attacks, APT, and lateral movements follow supply chain attacks. Supply chain attackers are increased to target the supply chain and exploit vulnerabilities through vendors and suppliers to gain access through organizational systems (Abdali et al., 2024). These attacks have a wide range of impacts, potentially analyzing multiple organizational customers. Advanced Persistent Threats: APT is characterized by stealth, persistence, and the ability to analyze the undetected for an extended period; it allows attackers to move through laterally within a network and steal sensitive data. Lateral movement attackers use various techniques to help gain initial access, which makes it difficult to contain the breach. Here also explained the AI driven threats as AI-power attacks, Deep fakes, Exploitation speed, insider threats, IoT device exploits, data breaches, injection attacks, and driven by compromise (Hong et al., 2025).

8.3 Collaboration Across Industries for LLM Security

To increase the LLM integration with various healthcare, finance, law, education, and customer services industries. The need for these security practices and models is robust and provides the possible information through prompt injection, data leakages, model hallucination, and adversarial manipulations. It is essential to develop effective security strategies to analyze the threats related to cross-cutting means, which affect multiple sectors and collaboration across various industries. To join the task, the efforts are allowed through sharing the threat of intelligence, which is enabled through industries learned from other experiences. Various organizations are co-investing in the robustness of the research-investing model, auditing tool, and secured deployment framework by pooling resources. Additionally, it is the way the development should share regular guidelines among the ethical frameworks and balance innovation followed by safety to ensure LLM deployed with responsibility among the sectors (Esposito et al., 2024).

8.4 Role of Policy and Governance in Securing LLMs

Policies are essential in securing the LLM and providing various structural foundations. It's also necessary for guides to take responsibility through development, deployment, and oversights. LLM is more prevalent across sensitive sectors like government, finance, and law. The government frameworks help to define accountability and clarify the responsibility of LLM to operate through unpredictability. Which also established the standards of transparency, explainability, and access control (Ferdaus et al., 2024). At the national and international level, the government should take the beginning stage of AI regulations, which should include safe, secure, and ethical principles. Furthermore, collaborative governance ideas relate to civil society and most private organizations. They all help to develop a balanced framework to encourage while safeguarding harm. The policy and governances are not regularly taken mechanisms to build public trust, and also established global standards ensure the transformative power of LLM responsibility is secure (Jiao et al., 2024).

IX. CONCLUSION

9.1 Summary of Key Findings

Large language models (LLMs) such as GPT drive chatbots and virtual assistants that transform human-computer interaction have significant flaws. Prompt injection attacks where hostile users control the model's input to evade safety filters or induce undesired behavior cause great worry. A user might, for example, include secret instructions inside a prompt to let the model overlook prior rules or limits. Closely linked to this is jailbreaking, in which one creates inputs such that the model executes limited actions, usually using encoded, inverted, or creatively written commands. Data leaking is another crucial flaw whereby the model could unintentionally recollect and expose private information from earlier interactions or training data. This raises serious privacy issues, especially when the virtual assistant is interfaced with systems handling personal or confidential information. Adversarial examples are inputs intentionally designed to mislead or trick the model, which may also result in incorrect or detrimental outputs. In sensitive domains such as finance or healthcare, these can particularly cause harm. Further, pressing issues also involve impersonation and disinformation. While eloquent and persuasive, LLMs do not inherently distinguish between correct and incorrect information. As such, they can be used to generate false identities or post deceptive content, which may spread misinformation or make social engineering easier. Users are likely to accept the model's output at face value, even if wrong or speculative, due to the over-reliance on LLMs, worsening the issue. Insecure API connections can also leave systems vulnerable, especially if chatbots are connected to backend operations such as user databases or payment gateways. Attackers can utilize such connections to run destructive commands or gain illegal access. Another problem is the explainability of LLM behavior. Since the models are black-box, it's difficult to determine why they arrived at a certain response; this makes debugging and security updates even more stimulating Developers and organizations need to implement strong safety measures to reduce these hazards. Some of these are using person-in-the-loop training methods like RLHF (reinforcement learning with human feedback), adding input and output filters, doing regular red-teaming exercises, and teaching people about AI systems' limits. To maintain the reliability and safety of chatbots and virtual assistants, it is crucial to demonstrate how the AI behaves and restrict access to essential features.

9.2 The Future of LLM Security

In the ever-changing landscape of LLM security, the constant struggle between new attack methods and improved defenses will determine the future. Future studies should enhance LLMs' resilience through real-time threat detection, continuous learning-based defense mechanisms, and transparent, explainable artificial intelligence. Improving fine-tuning methods like RLHF and including secure frameworks like federated learning and differential privacy will minimize vulnerability to malicious inputs and data

leaks. Establishing industry-wide guidelines for safe deployment would also require close cooperation between lawmakers, security researchers, and AI developers. Ethical rules and legal systems must change with technology to guarantee responsible use and liability. In the future, A critical issue is making sure LLMs are safe, aligned with values of society, and capable of human-like communicating and reasoning. The future of LLM privacy hinges on a firm commitment to governance, transparency, and continuous assessment as much as technological innovation.

9.3 Recommendations for Future Research

Safety research on LLMs ought to continue with a focus on creating defense systems that are rigorous, agile, and rooted in ethical foundations. A key area is creating consistent and robust evaluation standards that can assess LLMs against a wide range of attack types, such as adversarial prompts, data poisoning, and privacy leaks. Consistency and comparisons between numerous models and defensive measures rely on such criteria. The architecture of generalizable and adaptive defense systems not limited to particular threats but can develop in tandem with emerging and advanced attack vectors, perhaps through ongoing education or dynamic threat analysis systems, should also be another priority. Increasing the interpretability and explainability of LLMs is essential for researchers and developers to understand model behavior, identify defects, and monitor manipulative interventions. To avoid model outputs from leaking confidential information, incorporating privacy-preserving strategies such as federated learning, secure multi-party computation, and differential privacy is important. Future studies should also investigate the creation of intelligent monitoring systems with real-time automatic detection and mitigating capability for suspicious or hostile input. An additional course of action has been suggested, such as the use of attack simulators and red-teaming settings. This will enable the early detection and correction of vulnerabilities before their exploitation in actual applications.

REFERENCES

[1] Abdali, S., Anarfi, R., Barberan, C. J., He, J., & Shayegani, E. (2024). Securing large language models: Threats, vulnerabilities and responsible practices. https://doi.org/10.48550/arXiv.2403.12503

[2] Acosta, H., Lee, S., Bae, H., Feng, C., Rowe, J., Glazewski, K., ... & C. Lester, J. (2024). Recognizing Multi-Party Epistemic Dialogue Acts During Collaborative Game-Based Learning Using Large Language Models. *International Journal of Artificial Intelligence in Education*, 1-25.

[3] Adiwardana, D., Luong, M. T., So, D. R., Hall, J., Fiedel, N., Thoppilan, R., ... & Le, Q. V. (2020). Towards a human-like open-domain chatbot. https://doi.org/10.48550/arXiv.2001.09977

[4] Ando, R., Takahashi, K., & Suzaki, K. (2012). Inter-domain Communication Protocol for Real-time File Access Monitor of Virtual Machine. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, 3*(1/2), 120-137

[5] Andraško, J., Mesarčík, M., & Hamuľák, O. (2021). The regulatory intersections between artificial intelligence, data protection and cyber security: challenges and opportunities for the EU legal framework. *AI & society*, 1-14.

[6] Anthis, J., Lum, K., Ekstrand, M., Feller, A., D'Amour, A., & Tan, C. (2024). The impossibility of fair LLMs. https://doi.org/10.48550/arXiv.2406.03198

[7] Arvisais-Anhalt, S., Gonias, S. L., & Murray, S. G. (2024). Establishing priorities for implementation of large language models in pathology and laboratory medicine. *Academic Pathology*, *11*(1), 100101. https://doi.org/10.1016/j.acpath.2023.100101

[8] Ayyamperumal, S. G., & Ge, L. (2024). Current state of LLM Risks and AI Guardrails. https://doi.org/10.48550/arXiv.2406.12934

[9] Barman, K. G., Wood, N., & Pawlowski, P. (2024). Beyond transparency and explainability: on the need for adequate and contextualized user guidelines for LLM use. *Ethics and Information Technology*, *26*(3), 47. https://doi.org/10.1007/s10676-024-09778-2

[10] Bathala, N. K., & Babu, G. R. (2024). Adversarial ATTACKS on Large Language Models (LLMS) in Cybersecurity Applications: Detection, Mitigation, and Resilience Enhancement. *International Research Journal of Modernization in Engineering Technology and Science, 6*(10), 80-94.

[11] Biasin, E., Kamenjašević, E., & Ludvigsen, K. R. (2024). Cybersecurity of AI medical devices: risks, legislation, and challenges. *Research Handbook on Health, AI and the Law*, 57-74.

[12] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, *33*, 1877-1901.

[13] Chen, Y. J., & Madisetti, V. K. (2024). Information Security, Ethics, and Integrity in LLM Agent Interaction. *Journal of Information Security*, *16*(1), 184-196.

[14] Chen, Y., Cui, M., Wang, D., Cao, Y., Yang, P., Jiang, B., ... & Liu, B. (2024). A survey of large language models for cyber threat detection. *Computers & Security*, 104016. https://doi.org/10.1016/j.cose.2024.104016

[15] Chernyavskiy, A., Ilvovsky, D., & Nakov, P. (2021, September). Transformers: "the end of history" for natural language processing?. In *Joint European conference on machine learning and knowledge discovery in databases* (pp. 677-693). Cham: Springer International Publishing.

[16] Chowdhary, K., & Chowdhary, K. R. (2020). Natural language processing. *Fundamentals of artificial intelligence,* 603-649.

[17] Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., ... & Wei, J. (2024). Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, *25*(70), 1-53.

[18] Covington, N. V., & Vruwink, O. (2024). ChatGPT in Undergraduate Education: Performance of GPT-3.5 and Identification of AI-Generated Text in Introductory Neuroscience. *International Journal of Artificial Intelligence in Education*, 1-24.

[19] Dai, S., Xu, C., Xu, S., Pang, L., Dong, Z., & Xu, J. (2024, August). Bias and unfairness in information retrieval systems: New challenges in the llm era. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 6437-6447).

[20] Das, B. C., Amini, M. H., & Wu, Y. (2025). Security and privacy challenges of large language models: A survey. *ACM Computing Surveys, 57*(6), 1-39.

[21] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019, June). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)* (pp. 4171-4186).

[22] Dhruvitkumar, V. T. (2024). Ethical and Legal Issues of AI-based Health Cybersecurity.

[23] Donkor, K., & Zhao, Z. (2024). The Impact of Digital Transformation on Business Models: A Study of Industry Disruption. *Global Perspectives in Management, 2*(3), 1-12.

[24] Esmradi, A., Yip, D. W., & Chan, C. F. (2023, November). A comprehensive survey of attack techniques, implementation, and mitigation strategies in large language models. In *International Conference on Ubiquitous Security* (pp. 76-95). Singapore: Springer Nature Singapore.

[25] Esposito, M., Palagiano, F., Lenarduzzi, V., & Taibi, D. (2024). On Large Language Models in Mission-Critical IT Governance: Are We Ready Yet?. https://doi.org/10.48550/arXiv.2412.11698

[26] Ferdaus, M. M., Abdelguerfi, M., Ioup, E., Niles, K. N., Pathak, K., & Sloan, S. (2024). Towards trustworthy ai: A review of ethical and robust large language models. https://doi.org/10.48550/arXiv.2407.13934

[27] Freiberger, V., & Buchmann, E. (2024). Fair balancing? evaluating llm-based privacy policy ethics assessments. In *Proceedings of the 3rd European Workshop on Algorithmic Fairness (EWAF'24)*.

[28] Gan, Y., Yang, Y., Ma, Z., He, P., Zeng, R., Wang, Y., ... & Ji, S. (2024). Navigating the risks: A survey of security, privacy, and ethics threats in llm-based agents. https://doi.org/10.48550/arXiv.2411.09523

[29] Hassanin, M., & Moustafa, N. (2024). A comprehensive overview of large language models (llms) for cyber defences: Opportunities and directions. https://doi.org/10.48550/arXiv.2405.14487

[30] Haurogné, J., Basheer, N., & Islam, S. (2024). Advanced Vulnerability Detection Using Llm with Transparency Obligation Practice Towards Trustworthy Ai. https://doi.org/10.1016/j.mlwa.2024.100598

[31] He, F., Zhu, T., Ye, D., Liu, B., Zhou, W., & Yu, P. S. (2024). The emerged security and privacy of llm agent: A survey with case studies. https://doi.org/10.48550/arXiv.2407.19354

[32] Hemasree, V., & Kumar, K. S. (2022). Facial Skin Texture and Distributed Dynamic Kernel Support Vector Machine (DDKSVM) Classifier for Age Estimation in Facial Wrinkles. *Journal of Internet Services and Information Security, 12*(4), 84-101. https://doi.org/10.58346/JISIS.2022.I4.006

[33] Henkel, O., Hills, L., Roberts, B., & McGrane, J. (2024). Can LLMs Grade Open Response reading comprehension questions? An empirical study using the ROARs dataset. *International journal of artificial intelligence in education,* 1–26.

[34] Hong, Y., Wu, J., & Guan, X. (2025). A survey of joint security-safety for function, information and human in industry 5.0. *Security and Safety*, *4*, 2024014.

[35] Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., ... & Gelly, S. (2019, May). Parameter-efficient transfer learning for NLP. In *International conference on machine learning* (pp. 2790-2799). PMLR.

[36] Iyer, R., & Deshpande, N. (2024). Nanotechnology and their Applications in Chiral and Achiral Separating Mechanisms. *Engineering Perspectives in Filtration and Separation, 2*(4), 7-13.

[37] Jaber, H. A., Younis, H. S., Jiheel, W. R., Abdullah, R. M., & Hasan, S. A. (2025). Genetic Evaluation Study of Fava Bean Vicia Faba L. Under the Influence of the Transfer and Diagnosis of the Bean Yellow Mosaic Virus in Several Areas of Kirkuk Governorate. *Natural and Engineering Sciences, 10*(1), 151-161. https://doi.org/10.28978/nesciences.1642299

[38] Jiao, J., Afroogh, S., Xu, Y., & Phillips, C. (2024). Navigating llm ethics: Advancements, challenges, and future directions. https://doi.org/10.48550/arXiv.2406.18841

[39] Jing, H., Wei, W., Zhou, C., & He, X. (2021, June). An artificial intelligence security framework. In *Journal of Physics: Conference Series* (Vol. 1948, No. 1, p. 012004). IOP Publishing. https://doi.org/10.1088/1742-6596/1948/1/012004

[40] Johri, P., Khatri, S. K., Al-Taani, A. T., Sabharwal, M., Suvanov, S., & Kumar, A. (2021). Natural language processing: History, evolution, application, and future work. In *Proceedings of 3rd International Conference on Computing Informatics and Networks: ICCIN 2020* (pp. 365-375). Springer Singapore.

[41] Judge, B., Nitzberg, M., & Russell, S. (2025). When code isn't law: rethinking regulation for artificial intelligence. *Policy and Society, 44*(1), 85-97.

[42] Karimov, Z., & Bobur, R. (2024). Development of a Food Safety Monitoring System Using IOT Sensors and Data Analytics. *Clinical Journal for Medicine, Health and Pharmacy, 2*(1), 19-29

[43] Katulić, T. (2020). Towards the trustworthy AI: Insights from the regulations on data protection and information security. *Medijska istraživanja: znanstveno-stručni časopis za novinarstvo i medije*, *26*(2), 9-28.

[44] Kavibharathi, S., Lakshmi Priyankaa, S., Kaviya, M. S., & Vasanthi, S. (2021). Live Chat Analysis Using Machine Learning. *International Academic Journal of Science and Engineering, 8*(1), 39–44. https://doi.org/10.9756/IAJSE/V8I1/IAJSE0805

[45] Kemp, R. (2018). Legal Aspects of Artificial Intelligence (v. 2.0). *Kemp IT Law. – 2016. https://www.kempitlaw.com/wp-content/uploads/2016/11/Legal-Aspects-of-AI-Kemp-IT-Law-v2. 0-Nov-2016-. pdf.*

[46] Krishnan, M., & Patel, A. (2023). Circular Economy Models for Plastic Waste Management in Urban Slums. *International Journal of SDG's Prospects and Breakthroughs, 1*(1), 1-3.

[47] Kumar, A., Joshi, P., Bala, A., Sudhakar Patil, P., Jang Bahadur Saini, D. K., & Joshi, K. (2023). Smart Transaction through an ATM Machine using Face Recognition. *Indian Journal of Information Sources and Services, 13*(2), 7–13. https://doi.org/10.51983/ijiss-2023.13.2.3752

[48] Kumar, A., Murthy, S. V., Singh, S., & Ragupathy, S. (2024). The ethics of interaction: Mitigating security threats in llms. https://doi.org/10.48550/arXiv.2401.12273

[49] Kumar, P. (2024). Adversarial attacks and defenses for large language models (LLMs): methods, frameworks & challenges. *International Journal of Multimedia Information Retrieval*, *13*(3), 26. https://doi.org/10.1007/s13735-024-00334-8

[50] Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., ... & Zettlemoyer, L. (2019). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. https://doi.org/10.48550/arXiv.1910.13461

[51] Lieber, O., Sharir, O., Lenz, B., & Shoham, Y. (2021). Jurassic-1: Technical details and evaluation. *White Paper. AI21 Labs*, *1*(9), 1-17.

[52] Liu, F., Jiang, J., Lu, Y., Huang, Z., & Jiang, J. (2025). The ethical security of large language models: A systematic review. *Frontiers of Engineering Management*, 1-13.

[53] Maramreddy, Y. R., & Muppavaram, K. (2024). Detecting and Mitigating Data Poisoning Attacks in Machine Learning: A Weighted Average Approach. *Engineering, Technology & Applied Science Research, 14*(4), 15505-15509.

[54] Mehta, P., & Malhotra, K. (2024). Natural Language Processing for Automated Extraction of Medical Terms in Electronic Health Records. *Global Journal of Medical Terminology Research and Informatics, 2*(2), 1-4.

[55] Morris, W., Holmes, L., Choi, J. S., & Crossley, S. (2024). Automated scoring of constructed response items in math assessment using large language models. *International journal of artificial intelligence in education,* 1–28.

[56] Motlagh, F. N., Hajizadeh, M., Majd, M., Najafi, P., Cheng, F., & Meinel, C. (2024). Large language models in cybersecurity: State-of-the-art. https://doi.org/10.48550/arXiv.2402.00891

[57] Narayan, A., & Balasubramanian, K. (2024). Modeling Fouling Behavior in Membrane Filtration of High-Fat Food Emulsions. *Engineering Perspectives in Filtration and Separation, 2*(1), 9-12.

[58] Neupane, S., Mitra, S., Fernandez, I. A., Saha, S., Mittal, S., Chen, J., ... & Rahimi, S. (2024). Security considerations in ai-robotics: A survey of current methods, challenges, and opportunities. *IEEE Access, 12,* 22072-22097.

[59] Norberg, K. A., Almoubayyed, H., De Ley, L., Murphy, A., Weldon, K., & Ritter, S. (2024). Rewriting Content with GPT-4 to Support Emerging Readers in Adaptive Mathematics Software. *International Journal of Artificial Intelligence in Education*, 1-40.

[60] O'Sullivan, S., Nevejans, N., Allen, C., Blyth, A., Leonard, S., Pagallo, U., ... & Ashrafian, H. (2019). Legal, regulatory, and ethical frameworks for development of standards in artificial intelligence (AI) and autonomous robotic surgery. *The international journal of medical robotics and computer assisted surgery*, *15*(1), e1968. https://doi.org/10.1002/rcs.1968

[61] Paracha, A., Arshad, J., Farah, M. B., & Ismail, K. (2024). Machine learning security and privacy: a review of threats and countermeasures. *EURASIP Journal on Information Security*, *2024*(1), 10. https://doi.org/10.1186/s13635-024-00158-3

[62] Pearce, H., Ahmad, B., Tan, B., Dolan-Gavitt, B., & Karri, R. (2025). Asleep at the keyboard? assessing the security of github copilot's code contributions. *Communications of the ACM, 68*(2), 96-105.

[63] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2227–2237.

[64] Radiya-Dixit, E., Hong, S., Carlini, N., & Tramer, F. (2021). Data poisoning won't save you from facial recognition. https://doi.org/10.48550/arXiv.2106.14851

[65] Rae, J. W., Borgeaud, S., Cai, T., Millican, K., Hoffmann, J., Song, F., ... & Irving, G. (2021). Scaling language models: Methods, analysis & insights from training gopher. https://doi.org/10.48550/arXiv.2112.11446

[66] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, *21*(140), 1-67.

[67] Rathod, V., Nabavirazavi, S., Zad, S., & Iyengar, S. S. (2025, January). Privacy and security challenges in large language models. In *2025 IEEE 15th Annual Computing and Communication Workshop and Conference (CCWC)* (pp. 00746-00752). IEEE.

[68] Sakib, M. N., Islam, M. A., Pathak, R., & Arifin, M. M. (2024, September). Risks, Causes, and Mitigations of Widespread Deployments of Large Language Models (LLMs): A Survey. In *2024 2nd International Conference on Artificial Intelligence, Blockchain, and Internet of Things (AIBThings)* (pp. 1-7). IEEE.

[69] Sarker, I. H. (2024). LLM potentiality and awareness: a position paper from the perspective of trustworthy and responsible AI modeling. *Discover Artificial Intelligence*, *4*(1), 40. https://doi.org/10.1007/s44163-024-00129-0

[70] Singh, A. (2025). Evaluating the transparency and explainability of llm-based educational systems.

[71] Srinivasan, S., Mahbub, M., & Sadovnik, A. (2024). Advancing NLP Security by Leveraging LLMs as Adversarial Engines. https://doi.org/10.48550/arXiv.2410.18215

[72] Subramanian, R. (2017). Emergent AI, social robots and the law: Security, privacy and policy issues. *Subramanian, Ramesh (2017)" Emergent AI, Social Robots and the Law: Security, Privacy and Policy Issues," Journal of International, Technology and Information Management*, *26*(3).

[73] Sumartono, E., Harliyanto, R., Situmeang, S. M. T., Siagian, D. S., & Septaria, E. (2024). The Legal Implications of Data Privacy Laws, Cybersecurity Regulations, and AI Ethics in a Digital Society. *The Journal of Academic Science*, *1*(2), 103-110.

[74] Sun, Y., Wang, S., Feng, S., Ding, S., Pang, C., Shang, J., ... & Wang, H. (2021). Ernie 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation. https://doi.org/10.48550/arXiv.2107.02137

[75] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.

[76] Wang, A., Pruksachatkun, Y., Nangia, N., Singh, A., Michael, J., Hill, F., ... & Bowman, S. (2019). Superglue: A stickier benchmark for general-purpose language understanding systems. Advances in neural information processing systems, 32.

[77] Wei, Z., Sun, J., Zhang, Z., Zhang, X., Li, M., & Hou, Z. (2024). LLM-SmartAudit: Advanced Smart Contract Vulnerability Detection. https://doi.org/10.48550/arXiv.2410.09381

[78] Workshop, B., Scao, T. L., Fan, A., Akiki, C., Pavlick, E., Ilić, S., ... & Bari, M. S. (2022). Bloom: A 176b-parameter open-access multilingual language model. https://doi.org/10.48550/arXiv.2211.05100

[79] Wu, S., Zhao, X., Yu, T., Zhang, R., Shen, C., Liu, H., ... & Zhang, X. (2021). Yuan 1.0: Large-scale pre-trained language model in zero-shot and few-shot learning. https://doi.org/10.48550/arXiv.2110.04725

[80] Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., ... & Raffel, C. (2020). mT5: A massively multilingual pre-trained text-to-text transformer. https://doi.org/10.48550/arXiv.2010.11934

[81] y Arcas, B. A. (2022). Do large language models understand us?. *Daedalus*, *151*(2), 183-197. https://doi.org/10.1162/daed_a_01909

[82] Zhang, F., Li, C., Henkel, O., Xing, W., Baral, S., Heffernan, N., & Li, H. (2024). Math-LLMs: AI cyberinfrastructure with pre-trained transformers for math education. *International Journal of Artificial Intelligence in Education,* 1–24.

[83] Zhang, Z., Gu, Y., Han, X., Chen, S., Xiao, C., Sun, Z., ... & Sun, M. (2021). Cpm-2: Large-scale cost-effective pre-trained language models. https://doi.org/10.48550/arXiv.2106.10715