***Ethics and limitation discussion***

***What are the ethical considerations in the development of Large Language Models (LLMs)?***

Fairness, accountability, openness, privacy, and the possibility of bias are the main ethical issues in LLM development. Concerns are raised by bias in particular because it can result in skewed outputs that disproportionately affect particular demographic groups. Making sure that LLMs are used appropriately and do not reinforce negative societal stereotypes requires addressing these issues.

***Limitations and challenges of LLMs***

Large language models might give us the impression that they understand meaning and can respond to it accurately. However, they remain a technological tool and as such, large language models face a variety of challenges.

**Hallucinations:** A hallucination is when a LLM produces an output that is false, or that does not match the user's intent. For example, claiming that it is human, that it has emotions, or that it is in love with the user. Because large language models predict the next syntactically correct word or phrase, they can't wholly interpret human meaning. The result can sometimes be what is referred to as a "hallucination."

**Security:** Large language models present important security risks when not managed or surveyed properly. They can leak people's private information, participate in phishing scams, and produce spam. Users with malicious intent can reprogram AI to their ideologies or biases, and contribute to the spread of misinformation. The repercussions can be devastating on a global scale.

**Bias:** The data used to train language models will affect the outputs a given model produces. As such, if the data represents a single demographic, or lacks diversity, the outputs produced by the large language model will also lack diversity.

**Consent:** Large language models are trained on trillions of datasets — some of which might not have been obtained consensually. When scraping data from the internet, large language models have been known to ignore copyright licenses, plagiarize written content, and repurpose proprietary content without getting permission from the original owners or artists. When it produces results, there is no way to track data lineage, and often no credit is given to the creators, which can expose users to copyright infringement issues.

They might also scrape personal data, like names of subjects or photographers from the descriptions of photos, which can compromise **privacy**. LLMs have already run into lawsuits, including a prominent one by Getty Images3, for violating intellectual property.

**Scaling:** It can be difficult and time- and resource-consuming to scale and maintain large language models.

**Deployment:**Deploying large language models requires deep learning, a transformer model, distributed software and hardware, and overall technical expertise

***Bais in large language models***

Current LLMs are initially developed through a pre-training phase, where extensive data is fed into the model. This phase is essential in shaping the model’s ability to comprehend the complexities and abnormalities of natural language. After the pre-training phase, models are fine-tuned through instruction tuning, human preference alignment, and domain knowledge augmentation for enhanced reasoning and to cater to specific tasks and functions.

Biases can arise during both phases as the model learns undesired behaviors and information. Indeed, if an information source contains inaccurate content or if a data collection lacks diversity, a model can translate these biases into the text and content it generates for the user.  In short, bias in LLMs typically comes from the data that they were trained on, meaning the outputs of LLMs can reflect human and societal biases despite not being sentimental

Bias remains a significant concern when using AI in hiring. Language models can unintentionally reflect historical patterns of discrimination present in the training data. For example, they may over-prioritize candidates with specific educational backgrounds, work formats, or language styles that are statistically more common in certain regions or demographics.

In the prompts used, care was taken to neutralize such bias by focusing on **skills**, **outcomes**, and **collaboration examples** rather than personal details or formatting. However, the risk of implicit bias still exists, particularly if prompts are vague or overly dependent on keyword matching, which can unfairly penalize candidates who describe their experience differently.

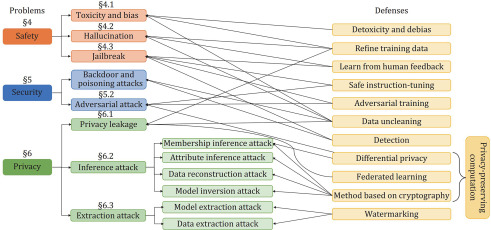
***Safety and security of large language models***

Safety refers to the model’s inherent ability to function correctly and consistently without unintended behaviors or errors in the absence of external threats. It ensures that the model adheres to ethical guidelines, avoids harmful outputs, and operates within its intended design parameters.

Security in LLMs addresses the model’s vulnerability to intentional attacks by adversaries. It focuses on the model’s resilience and robustness to manipulations that could lead to incorrect, misleading, or malicious outputs.

Privacy concerns the protection of sensitive information, including the model’s training data and parameters and users’ personal information. It aims to prevent unauthorized access, use, or disclosure of private information.

Building on these definitions, we provide a comprehensive overview of the vulnerabilities and defense mechanisms associated with LLMs. , we categorize current research into three main areas: Safety, security, and privacy. Under each category, we identify specific security issues and outline corresponding defensive measures to mitigate these vulnerabilities.



. Overview of safety, security, and privacy issues and their defence methods.

 presents the structure of this study, outlining the safety, security, and privacy threats faced by LLMs .

***Responsible prompt engineering***

The successful operation of large language models (LLMs) heavily depends on using prompt engineering properly.

Users can achieve better performance from large language models through well-structured prompts that guide the

models to produce accurate, contextually valid and insightful outputs. The improvement of prompt effectiveness

depends on three main techniques: structured prompting as well as iterative refinement and prompt tuning [1]. These

techniques minimize ambiguities while improving the coherence of AI output while maintaining consistent results.

Structured prompting proves to be one of the most valuable techniques which require designers to prepare prompts

that define specific roles and contexts and impose constraints. Role-based prompting requires the model to perform

like a cybersecurity expert through instructions such as "Act as a cybersecurity expert" while also requesting it to

provide legal advice through "Provide legal advice based on case law." The approach strengthens domain-related

precision through expert knowledge alignment which guides model response patterns [1]. Contextual scaffolding

enables prompt developers to provide sufficient background information which helps the model generate detailed

responses [2]. This directive tells the model to produce a 150-word summary which concentrates on research

methodology and key findings. The response refinement method of constraint-based prompting allows users to

define response guidelines through parameters which determine word count and output design among other elements

(such as bullet-point vs paragraph formats).

The process of iterative refinement requires regular prompt modifications through previous model outputs to

enhance response quality. The integration of feedback loops enables users to modify their input queries

automatically for improved results. The responses become clearer when prompt rewording includes targeted

keywords or when additional contextual information is provided for ambiguous initial questions. Chain-of-thought

prompting serves as a refinement method that guides the model to undertake systematic thinking processes. The

model demonstrates great value for mathematical problem-solving by splitting complex problems into step-by-step

logical sequences [2].

Performance enhancement through machine learning occurs through two optimization approaches which include

manual prompt refinement in combination with prompt tuning methods. Through few-shot and zero-shot learning

strategies LLMs can better understand different domains using only limited training examples. The LLM receives

instructions such as "Translate the following sentence into French: 'The weather is nice today'" (few-shot) to learn

the expected output format. Embeddings coupled with

meta prompting enable dynamic model interactions that let

the system modify its responses through learning from preceding user queries