Exploring the Potential of Large Language Models for Improving Digital Forensic Investigation Efficiency

Akila Wickramasekara[1], Frank Breitinger[2,3], Mark Scanlon[1]

[1]*Forensics and Security Research Group, School of Computer Science, University College Dublin, Belfield, Dublin 4, Ireland*
[2]*School of Criminal Justice, University of Lausanne, Lausanne, Switzerland*
[3]*Present affiliation: Institute of Computer Science, University of Augsburg, Augsburg, Germany*

Abstract

The ever-increasing workload of digital forensic labs raises concerns about law enforcement's ability to conduct both cyber-related and non-cyber-related investigations promptly. Consequently, this article explores the potential and usefulness of integrating Large Language Models (LLMs) into digital forensic investigations to address challenges such as bias, explainability, censorship, resource-intensive infrastructure, and ethical and legal considerations. A comprehensive literature review is carried out, encompassing existing digital forensic models, tools, LLMs, deep learning techniques, and the use of LLMs in investigations. The review identifies current challenges within existing digital forensic processes and explores both the obstacles and the possibilities of incorporating LLMs. In conclusion, the study states that the adoption of LLMs in digital forensics, with appropriate constraints, has the potential to improve investigation efficiency, improve traceability, and alleviate the technical and judicial barriers faced by law enforcement entities.

Keywords:
Digital Forensics, Large Language Models, LLM, Investigative Process, Challenges

1. Introduction

With the widespread growth of information and communication technology (ICT) and information systems, cybercrimes have seen a significant increase in recent years [1][1]. As a further compounding factor, the number of "traditional" police investigations that include digital evidence is also increasing [2]. Addressing and investigating this volume of cases presents substantial challenges.

Generative AI (GenAI) and Large Language Models (LLMs) have become prominent topics of global discussion, prompting researchers to intensify their investigations by leveraging the capabilities of LLMs. The usage of LLMs within the scientific community experienced a rapid surge after 2022, notably with the advent of OpenAI's ChatGPT platform. In a relatively short period of time, this topic has attracted the attention of academia, industry, and the research community at large [3]. Simultaneously, researchers are exploring the potential of LLMs in various domains and assessing their impact on the future of science and society. This inquiry also includes an examination of the potential harmfulness associated with the deployment of LLMs [4, 5]. In other words, the use of LLMs in various tasks can be a double-edged sword, necessitating careful consideration depending on the specific situations and contexts.

Given the rapidly evolving landscape of LLMs, it is prudent to look into various types and their unique capabilities. A nuanced understanding of the strengths and characteristics of different LLMs can contribute to more informed and effective applications within the dynamic field of digital forensics (DF).

This paper reviews recent advances in the application of LLMs within digital forensics, focussing on established models, methods, and key challenges. By examining contemporary studies from 2019 onwards, the survey highlights core areas, such as automation, investigative methods, and efficiency improvements facilitated by LLMs. In addition, it explores the literature that addresses challenges in both DF and LLMs, covering limitations, ethical considerations, and forensic-specific risks. This comprehensive review synthesises current insights and emerging trends, offering a foundation for understanding the potential and limitations of LLMs in DF contexts.

In light of fast-paced advancements and the recent explosion in LLM-focused research, a substantial influx of LLM-focused research papers has occurred since the launch of ChatGPT in late 2022. Due to this fast pace, many seminal research articles exist solely as preprints on preprint services, e.g., arXiv. To give two examples, the initial papers for GPT-4 [6] and LLaMA [7] are only published on arXiv, but have garnered thousands of citations each. The recent release of DeepSeek-V3 in late 2024 and the corresponding technical reqport [8] is also only available on arXiv. Despite their preprint status, these articles offer essential insights critical for contemporary research and dialogue within the domain, making their incorporation into this article necessary to provide the most up-to-date knowledge and perspectives.

The paper is structured as follows: Section 1.1 provides a

[1]https://go.crowdstrike.com/rs/281-OBQ-266/images/GlobalThreatReport2024.pdf

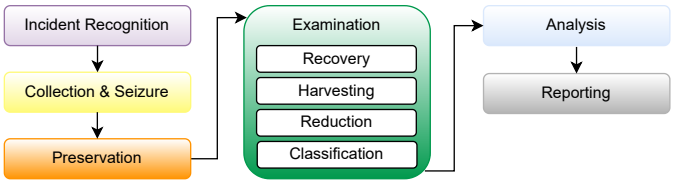

Figure 1: Traditional digital forensic process model [12]

comprehensive background for the review, delving into existing DF process models, the challenges inherent in DF, and a detailed overview of the current work conducted with the use of LLMs within DF. In Section 2 the paper delves into the realm of Natural Language Processing (NLP), elucidating the working principles of LLMs, their architectural foundations, and the specifics surrounding specially trained LLMs. Section 3 provides an in-depth review that focusses on the capabilities and benchmark information of LLMs trained for coding tasks, as well as those tailored for vision assistance. Section 3 explores the synergy between DF and LLMs, detailing how LLMs can be effectively employed in each phase of the DF process model. Finally, in Section 6, the paper summarises the future challenges associated with integrating LLMs with automated agents within the DF domain. The conclusion outlines potential avenues for future research and development, shedding light on the path of future DF investigations employing LLMs. The discussion covers not only the potential negative impacts but also the practical difficulties and risks in real world environments.

1.1. Digital Forensic Context

DF is a process for identifying, preserving, analysing, and documenting digitally recorded data, which originate in electronic devices such as computers, servers, smartphones, and IoT devices [9]. This exercise is required in most criminal cases. Data collected in this process are kept unchanged and safe for presentation in a court case or to support future investigations conducted by law enforcement agencies [10].

1.1.1. Digital Forensic Process Models

DF process models consist of a series of activities that help standardise the investigative process [11] and outline the phases; collection, preservation of evidence, examination or analysis, and reporting.

DF encompasses various subdisciplines such as computer forensics, mobile device forensics, memory forensics, network forensics, and cloud forensics, each employing distinct processes reflected in a plethora of models within the literature [13, 14]. These models often share phases but differ in their focus and execution. For example, Al-Dhaqm et al. [15] proposed a mobile forensic model that adds a preparatory phase and bifurcates the analysis stage into examination and analysis phases. To accommodate the complexities of computer, network, cloud, and smart device forensics, Lutui [16] introduced a multidisciplinary model that requires diverse skills for effective investigation, ranging from incident detection to evidence storage.

Casey's model, as shown in Figure 1, includes phases such as incident recognition, evidence collection, preservation, and presentation, with the examination phase detailed in recovery, harvesting, reduction, and classification [12]. During incident recognition, the focus is on identifying the incident itself, possible evidentiary sources, and expected digital evidence types, as well as delineating the scope of the ensuing investigation. Conversely, investigators systematically acquire pertinent evidence from various sources encompassing computers, smartphones, storage media, and networks during collection and seizure. Preservation is of paramount importance in upholding the integrity of evidence, necessitating specific and accurate measures to ensure the unmodified condition of the collected data throughout the investigative process. The overarching objective remains the meticulous safeguarding of evidence integrity.

The subsequent phase entails examination, in which analysts rigorously scrutinise the gathered data to extract pertinent information. This endeavour may involve the use of various forensic hardware and software tools and techniques. The examination process includes the interpretation of the information extracted to draw conclusive inferences about the events under scrutiny. This phase often demands a profound understanding of both the technology employed and the context surrounding the evidence.

Next is the reporting phase, where the findings derived from the analysis are presented systematically in a format suitable for legal adjudication. This may involve preparing comprehensive reports and providing expert testimony in a court of law. This model emphasises the critical nature of maintaining the integrity and provenance of the evidence and the requirement for expert analysis to extract and interpret pertinent information. This culminates in a report suitable for legal scrutiny. In particular, the analysis or examination stage is crucial in all models, demanding specialised knowledge in the relevant DF area [17, 18].

The advent of cloud computing has led to the Digital Forensic as a Service (DFaaS) model by van Baar et al., which integrates evidence preservation and analysis into an automated and secure software service, marking a significant evolution in forensic methodologies [20, 11, 21].

1.1.2. Existing Challenges in Digital Forensics

DF is an evolving field, yet the literature highlights that it still undergoes changes to address ongoing challenges and advancements. Dubey et al. [22] assert that DF faces key challenges, including the complexity of data and its volume, a lack of standardisation, inadequacies in the power of existing tools to support investigations, and issues related to timelines.

In addition to the previously mentioned challenges, other issues persist including scope creep in cases due to complexity and the vast data involved, selecting and prioritising the right set of evidence, and efficiently allocating time and investigators for the chosen evidence [23]. Koper et al. [24] focus on a number of these issues from the investigator's perspective, including challenges in adapting to a system, unexpected time sinks, and frustrations among officers arising from expected operational timelines and the adoption of complex systems. The contemporary landscape of forensic science is characterised by a

shortage of proficient agents, exacerbated by the swiftly evolving standards, practices, tools, and techniques within the field. Moreover, the predominant emphasis of law enforcement roles on fieldwork, as opposed to dedicated DF duties, has further contributed to the scarcity of adept human expertise in this domain [25].

Automating the DF process using existing technology appears to be a promising solution to address issues related to time management and effectiveness [26]. However, an ongoing challenge revolves around measuring the accuracy of investigations and ensuring the verification of the automated process. This aspect remains an open area that requires more attention and resolution [27].

1.1.3. Existing Work With LLMs in Digital Forensics

Scanlon et al. [13] analysed using ChatGPT for DF. In their assessment, the authors evaluated the programming, incident narration, keyword list creation, and DF teaching abilities of ChatGPT. Their conclusion highlighted that while ChatGPT exhibited some hallucinations in the output results, it still serves as an effective assistant for code generation. Wickramasekara et al. [28] introduced the AutoDFBench benchmarking framework, and corresponding score, to evaluate code generation for DF specific tasks against the tests and datasets used as part of NIST's Computer Forensics Tool Testing Program (CFTT)[2].

Timeline reconstruction helps investigators deduce the chronological "story" of an event. In line with timeline regeneration, Silalahi et al. [29] proposed a method to detect anomalies in a drone flight by employing sentiment analysis with the assistance of a pre-trained LLM. Their approach successfully discerned the differences between normal and abnormal events with an accuracy of 92.5%.

Hansken, a DFaaS platform created by the Netherlands Forensic Institute, is designed to help investigators handle evidence and conduct investigations more efficiently [20]. ChatGPT has been used as an assistance for the Hansken DFaaS system using its bespoke query language, contributing to streamlined processes and improved support for investigators. In these experiments by Henseler and van Beek [30], it was tasked with analysing evidence using Hansken's trace model. This work demonstrated the potential for ChatGPT in helping with analytical aspects of investigations, highlighting its ability to process and interpret evidence data.

While DF is the main focus of this paper, it is important to recognise the broader application of LLMs in adjacent areas within cybersecurity, many of which overlap with DF. LLMs are proving to be valuable tools in fields such as malware analysis, security log analysis, code security reviews, and intrusion detection areas that bridge the gap between cybersecurity and DF [31, 32, 33]. In malware analysis, LLMs can identify patterns in malicious code, while in log analysis, they assist in detecting anomalies across large datasets, thereby improving response times. In code-related security reviews, LLMs like GRACE have demonstrated the ability to identify vulnerabilities in software, achieving a detection rate of 28.65% of vulnerabilities [34]. These applications contribute to DF investigations by improving and improving evidence collection and analysis.

[2] `https://www.nist.gov/itl/ssd/software-quality-group/computer-forensics-tool-testing-program-cftt`

2. Background of Large Language Models

This section explores LLMs, concentrating on three principal aspects. Initially, it explores the architecture of LLMs, detailing their design and function. Then it assesses the usability of LLMs, underscoring the features and capabilities that make them suitable for a wide range of tasks. Finally, it showcases the versatility of LLMs by discussing their applications across various fields, demonstrating their wide-reaching impact and the extensive scope of their applications.

2.1. Natural Language Processing

Popular LLMs such as Generative Pre-trained Transformer (GPT) [35], Language Model for Dialogue Applications (LaMDA) [36], Pathways Language Model (PaLM) [37], Bidirectional Encoder Representations from Transformers (BERT) [38], and Large Language Model Meta AI (LLaMA) [7] stem from advances in Natural Language Processing (NLP). NLP, which focusses on language-based tasks, uses traditional and deep learning models to enable applications such as language translation, text processing, and speech recognition [39].

Deep learning, a branch of machine learning, uses complex computational layers and adaptive weights to improve prediction accuracy, offering a more refined analysis than conventional machine learning [40]. It has excelled in image and speech recognition and natural language understanding, mimicking the decision-making process of the human brain through artificial neurons. These neurons form networks capable of intricate pattern recognition and data analysis. Central to deep learning are neural networks with multiple hidden layers that autonomously learn and extract features from data, bypassing the need for manual variable selection. This automatic feature extraction makes them exceptionally adept at handling complex tasks [41].

2.2. LLMs

An LLM is a language model that employs neural networks with billions of parameters, trained on extensive text data. These models are engineered to comprehend and generate human language. Fundamentally, they rely on multiple neural network architectures, enabling them to recognise the relationships between words and phrases within sentences [42, 43]. These architectures have been a transformative force in natural language processing. Its capability to excel across a diverse array of language-related tasks distinguishes it as a game-changer, in contrast to being tailored for a singular, specific task.

2.3. Architecture of LLMs

LLMs utilise deep learning, particularly neural networks, to process and produce human language. Fundamentally, a language model operates with letters or words, but since machine learning algorithms and neural networks require vector inputs, words are vectorised. Each word in the vocabulary is assigned a unique numerical value for input into neural networks. Through initial random weight assignments and subsequent backpropagation, words acquire numerical positions reflecting their semantic similarity, culminating in a word embedding model [44].

2.3.1. Word to Vectors

Word embeddings, as introduced by Mikolov et al. [45], entail precise and high-dimensional vector representations for words, particularly suited for extensive datasets comprising billions of text entries. The authors explored model architectures for word vectorisation, achieving substantial improvements in accuracy while requiring fewer computational resources and reduced training time [45]. In the realm of LLMs, the primary objective is to generate new text based on the extensive dataset on which it was trained. For this purpose, Vaswani et al. [46] introduced the transformer model, assisted by the word-to-vector model. This architecture incorporates a self-attention mechanism, as well as encoder and decoder processes, enabling the model to quickly and simultaneously focus on pertinent information.

2.3.2. Transformer Models

The transformer model initially aimed at machine translation, translating input words into another language, begins with word embedding, where input, termed tokens, are vectorised. Recognising word order is achieved through positional encoding, with two main techniques: absolute and relative. Absolute positional encoding assigns unique vectors to each position, enhancing the model's ability to recognise word placement and facilitate position-specific attention [47]. Relative positional encoding, on the other hand, calculates the relative positions of words by introducing a bias term that quantifies the distances between positions, improving the model's ability to understand the relationships between words within a sequence [47].

Self-attention, a core mechanism within the transformer, calculates the relationship among words in a sentence, allowing the model to assess the similarity of each word with others and generate unique representations for each [48]. The decoder mirrors the encoder's steps but uses different weights, starting with positional encoding and computing self-attention values to identify the sentence's initial translation word.

This transformer process, which leverages stacking self-attention and unique positional encoding, has significantly advanced NLP tasks, including machine translation, text generation, and summarisation, by executing these processes in parallel and optimising the weights of both the encoder and decoder [46, 49].

2.4. Specifically Trained LLMs

The transformer model and the self-attention mechanism have paved the way for researchers to train language models on trillions of tokens with billions of parameters. Several LLMs have been trained and harnessed, each designed with specific capabilities for diverse fields such as security [33], chemistry [50, 51], engineering [52], medicine [53], business [54], tourism [55], and language-related applications [53]. These models are used in tasks ranging from detecting security threats [33], analysing data and generating synthetic actions to teaching [56], code generation [52, 57], structured query generation [54, 58], planning [54], assisting in medical education [53], clinical decision-making [59], leveraging clinical settings [60], clinical validation [61], understanding general patterns and decision-making [54], bias detection [56], addressing ethical issues [62], language translations [63], question answering [56], information extraction [64], and business process automation [54], among others [53]. The fine-tuning and retraining capabilities of LLMs enable them to be adapted to specific tasks or behaviours in a predefined manner. Fine-tuning involves taking an already trained language model and retraining its existing weights and bias values using a new dataset specific to a particular domain. This process allows the LLM to be customised and refined for tasks beyond its original training, enhancing its applicability in specific contexts [65]. This process results in a new model that is more tailored and focused on the specified domain. In the existing literature, it is frequently observed that LLMs are fine-tuned with a particular emphasis on engineering and research-related fields. This targeted fine-tuning ensures that the model is adept at handling tasks and generating content specifically relevant to the intricacies of these domains [53].

2.5. Multi-modal Large Language Models (MLLMs)

Unlike traditional LLMs, which are trained on text data, Multi-modal Large Language Models (MLLMs) are designed to process and interpret image-based data alongside text. These models can extract and analyse information within images or videos, integrating visual and textual data to enhance comprehension and analysis [66]. The application of MLLMs is rapidly expanding across fields such as digital forensics, where they can be used to analyse images of documents or identify visual anomalies. Section 3.2 provides a discussion on the potential and diverse applications of MLLMs in various domains.

2.6. Large Action Models

While LLMs excel in text generation and processing, they struggle with complex task manipulation and operational control, especially when moving from language understanding to action execution. This limitation arises because their core design emphasises prediction and generation over direct task execution. To overcome this shortcoming, recent research has introduced innovative approaches, such as the Large Action Model (LAM) developed by the Rabbit research team. LAM extends the capabilities of LLMs by incorporating action-based operations[3]. These LAMs can mimic human routines such as

[3] https://www.rabbit.tech/research

scheduling meetings with given instructions, sending emails, ordering taxis, and handling complex tasks such as making reservations for a whole trip. In this approach, the base model is trained to comprehend sequences of human-provided actions and commands, allowing it to execute these actions and tasks accordingly. Similarly, Microsoft introduced the concept of Visualisation-of-Thought (VoT) aimed at integrating human cognitive abilities, specifically the creation of mental images, into the model [67]. Through this approach, it has been demonstrated that MLLMs excel in visual tasks, thereby enabling the extension of action capabilities within an LAM to any LLM. These advancements signify promising directions toward enhancing the practical applicability and versatility of language models across various domains.

3. Capabilities of Large Language Models

This section focusses on the abilities and capabilities of Language Model Models (LLMs) as outlined in Section 2.4. This section also discusses the currently available fine-tuned LLMs that exhibit potential for application in DF. Although considered too broad for this article, Zhao et al. [68] provide a detailed generic overview of LLMs, their operation, and how they are trained and fine-tuned.

3.1. Programming/Code Generation

The ability to generate source code within a specific context is a crucial skill inherent in a language model [69]. Xu et al. [70] conducted a systematic evaluation of six LLMs for code generation in 12 different programming languages. The benchmarking process employed the HumanEval benchmark, along with a tailored evaluation dataset designed to assess the functional correctness of the programs generated by an LLM [71]. The benchmark comprises a set of coding problems in which the model is tasked with generating Python functions. Each problem is accompanied by a prompt and a set of unit tests that verify whether the generated code produces the expected output. This facilitates a systematic evaluation to generate both syntactically correct and functionally accurate programs. Using this dataset, it is possible to measure performance on real-world coding tasks, as well as its ability to generate solutions that satisfy functional requirements.

The Mostly Basic Programming Problems (MBPP) is another benchmark comprising 974 programming tasks. It serves as a frequently used evaluation dataset for LLMs specialising in code-related tasks [72]. Several LLMs explicitly trained for code generation include Code LLaMA [73], CodeGen [74], StarCoder [75], PanGu-Coder [76], PanGu-Coder2 [76], WizardCoder [77], InCoder 6B [78], CodeGen-Mono 16B [74], Code-Davinci-001 [79], Code-Davinci-002 [79], PaLM-Coder-540B [37], CodeT5+ [80, 81], InstructCodeT5+ [82, 81], GPT-4 with Reflexion [83], CodeGeeX [84], AlphaCode [85], Santa-Coder [86], CodeFuse-13B [87], Codex [88], WaveCoder [89]. A higher value for both HumanEval and MBPP indicates greater precision in code generation for a given task. For detailed information, refer to Table 1, which presents the counts for HumanEval and MBPP, along with the trained parameter size for each LLM. A higher score for both HumanEval and MBPP indicates greater precision in code generation for a given task.

Table 1: Trained parameter count, HumanEval and MBPP scores for LLM based code generation (ordered by HumanEval score).

Model	Parameters	HumanEval	MBPP
o1-mini	100B	97.6	93.9
GPT-4 with Reflexion	1.76T*	91.0	77.1
DeepSeek-V3	671B	85.6	-
DeepSeek-V3-Base	671B	65.2	75.4
Code LLaMA	34B	62.2	61.2
PanGu-Coder2	15B	61.64	-
WizardCoder	15B	57.3	51.8
Code-Davinci-002 (GPT3.5)	175B	47.0	58.1
StarCoder	15.5B	40.8	49.5
Code-Davinci-001 (GPT3)	175B	39.0	51.8
PaLM-Coder	540B	36.0	47.0
InstructCodeT5+	16B	35.0	-
code-cushman-001	12B	33.5	45.9
CodeT5+	16B	30.9	-
CodeGen-MONO	16.1B	29.7	42.4
CodeGen	16.1B	29.28	35.28
Codex-12B	12B	28.81	-
PanGu-Coder	2.6B	27.78	23
Sanata-Coder	1.1B	18	35
AlphaCode	1.1B	17.1	-
InCoder 6B	6.7B	16.4	21.3

Table 1 presents the scores for HumanEval and MBPP for each of the code generation LLMs mentioned above, along with the trained parameter size for each LLM. Four generic LLMs, or Mixture-of-Experts (MoE) models, are also included in Table 1: o1-mini [90], GPT-4 with Reflexion, DeepSeek-V3 and DeepSeek-V3-Base [8]. These are included as these are the top 4 best performing models for HumanEval despite them being MoE models.

3.2. Vision Assistance

Traditional vision assistant systems face limitations in image processing or recognition, as they are typically trained on fixed types of datasets. However, with the emergence of LLMs, this paradigm has changed to the use of raw text as a source of supervision [92, 93]. Research on visual recognition language models is experiencing exponential growth, with the number of models exceeding 1,500 in 2023 [94]. Radford et al. [92] introduced a novel method called Contrastive Language Image Pre-training (CLIP). This method is efficient and capable of performing a wide range of tasks during pre-training. It enables a model to learn a shared representation space for both

* Estimated parameter count as value is not officially released [91].

images and text, facilitating a deeper understanding of the relationships between the two modalities. Ramesh et al. [95] proposes a model for text-to-image generation, capable of generating images as combinations derived from textual input or sentences. Moreover, with the model named Generating Images with Large Language Models (GILL), it becomes feasible to generate text, retrieve images, generate novel images, and interleave the results into coherent multimodal dialogues [96]. VisionLLM is a framework leveraging LLMs for diverse vision tasks with unified language instruction, demonstrating generality and flexibility [97]. It incorporates a language-guided image tokeniser and an LLM-based task decoder, capable of handling open-ended tasks based on provided language instructions [97].

Visual instruction tuning leverages language-only models, such as GPT-4, to generate multimodal language-image instruction following data. This data is then utilised to instruction-tune large multimodal models, such as Large Language and Vision Assistant (LLaVA) [98, 99, 6]. The open source LLaVA project introduces an end-to-end trained model, integrating a vision encoder with an LLM. Notably, LLaVA showcases multimodal chat capabilities. LLaVa has the capability to interact with images, provide detailed descriptions and respond to queries with a reported accuracy of 92.53% [98]. This shows its effectiveness in understanding and generating contextually relevant information on visual content [98]. MiniGPT-4 is an open-source, powerful visual instruction-tuned LLM, and it demonstrates versatility by generating stories and poems inspired by provided images and teaching users how to cook based on visual cues from food photos. This showcases its ability to understand and respond creatively to various visual stimuli [100].

Position-Enhanced Visual Instruction Tuning (PVIT) represents an extended version of Multimodal Large Language Models (MLLMs). It facilitates region-level encoding in an image, enabling the model to discern and identify information within specific regions [101]. This model enables users to interact with both the language and drawing the bounding boxes to indicate the area of interest within an image [101]. Other MLLMs, such as Visual ChatGPT [92], InternGPT [102], Flamingo [103], BLIP-2 [104], and Kosmos [105], are noted in the literature for their ability to assist users in visual-related information.

Video information is gaining prominence in vision assistance, and Zhao et al. [106] has introduced a novel approach to automatically narrate lengthy videos using LLMs. UniViLM is another language pre-trained model designed for both multimodal understanding and generation. It is capable of retrieving a video segment based on text descriptions, generating captions for given video clips, segmenting a video according to a provided text input, and performing multimodal sentiment analysis of a video segment [107]. VidIL and LLaViLo are additional MLLMs with similar capabilities, demonstrating proficiency in video classification and video-language operations such as video captioning, video question answering, video caption retrieval, and prediction of future video events [108, 109].

These MLLMs adhere to a shared task set, that includes visual question answering, visual captioning, visual commonsense reasoning, visual generation, multimodal affective computing, visual retrieval, vision language navigation, multimodal machine translation, visual question generation and visual dialoguing, as summarised in Table 2 [110, 111].

3.3. Conversation

Specific LLMs are trained explicitly for meaningful and coherent dialogues with humans. An example is Dialogue Generative Pre-trained Transformer (DialoGPT), a fine-tuned model trained on 174 million Reddit conversations [112]. DialoGPT exhibits the ability to provide human-like answers in tested conversations [112]. Dettmers et al. [113] introduced a fine-tuning mechanism for LLMs named Quantised Pre-trained Language Model into Low-Rank Adapters (QLoRA). This allows for the fine-tuning of large-parameter LLMs with low training costs. They introduced Guanaco, a fine-tuned LLM with 65 billion parameters, which achieved a performance level of 99.3%. Falcon-180B and Falcon-40B represent another set of open-source LLMs with 180 billion and 40 billion parameters. These models are trained to communicate in multiple languages, allowing users to engage in conversations in languages other than English [114]. To evaluate the accuracy of human-like dialogue systems, Ou et al. [115] proposed a dialogue evaluation benchmark named DialogBench, which consists of 12 dialogue tasks to assess the capabilities of LLMs. In their evaluation, they assessed 28 pre-trained and instruction-tuned LLMs, demonstrating that GPT-4, ChatGPT, and KwaiYii-13B-Chat emerged as the top three models for conversations in domains related to daily life and professional knowledge.

In DF chat conversations, the significance lies in facilitating non-technical investigators to elucidate terminologies and areas lacking understanding. This serves a dual purpose, acting as an interactive teacher to enhance comprehension in discussions [13].

3.4. Prompt Engineering

Achieving quality outputs from LLMs often relies on providing well-crafted, meaningful, and precise input queries, known as input prompts. However, even human-defined natural language instructions may not consistently yield the best results. Prompt engineering is a methodology that involves carefully defining and instructing LLMs to generate more accurate and desirable outputs. Through thoughtful refinement of input prompts, prompt engineering aims to enhance the performance and effectiveness of LLMs in generating output that align more closely with user expectations and requirements [116, 117]. This plays a crucial role in biasing LLMs toward specific domains or topics, enabling a more targeted and nuanced response. By carefully crafting prompts, users can guide LLMs to dive deeper into the nuances of their queries, leading to more accurate and relevant outputs. This approach enhances the model's responsiveness to specific areas of interest, allowing users to fine-tune and tailor their interactions with the LLM for more precise and meaningful outcomes. Prompt engineering with LLMs is employed in various sectors, including, but not limited to, medical, engineering, construction, and healthcare [118, 119].

Table 2: Capabilities of MLLMs trained for vision assistance

Capabilities		GILL	VisionLLM	GPT-4	LLaVa	MiniGPT-4	Visual ChatGPT	InternGPT with Husky	Flamingo	Kosmos	UniVL	VidIL
Visual question answering	Image	✓	✓	✓	✓	✓	✓	✓	✓	✓		
(Task of providing an answer to a visual input.)	Video										✓	✓
Visual captioning	Image	✓	✓	✓	✓	✓	✓	✓	✓	✓		
(Task of generate fitting visual descriptions.)	Video										✓	✓
Visual common-sense reasoning	Image	✓	✓	✓	✓	✓	✓	✓	✓	✓		
(Task of infer understanding from images or video clip.)	Video										✓	✓
Visual generation	Image	✓					✓	✓				
(Task of generating image or video from a given textual input.)	Video											
Multimodal affective computing	Image	✓	✓	✓	✓	✓	✓	✓	✓	✓		
(Task of automatically recognition of emotions and causes.)	Video											
Visual retrieval	Image	✓	✓				✓	✓				
(Task of language and vision understanding and retrieval.)	Video										✓	
Vision-language navigation	Image	✓					✓	✓				
(Task of navigation based on linguistic instructions.)	Video										✓	
Multimodal machine translation	Image		✓	✓	✓	✓	✓	✓	✓	✓		
(Task of translation from a video or an image)	Video											
Visual question generation	Image		✓	✓	✓	✓	✓	✓	✓	✓		
(Task of generating questions for given image or video)	Video											✓
Visual dialoguing	Image	✓		✓	✓	✓	✓	✓	✓	✓		
(Task of automating a conversation about a video or image)	Video											✓

ChainForge is an open source Graphical User Interface (GUI) tool developed specifically for prompt engineering and hypothesis testing derived from LLMs that can be used in the aforementioned fields to generate accurate and quick output [120].

Although prompt engineering is a necessity in generating the desired output from an LLM, the results can still be biassed based on the wording and phrasing provided by the user. Since the effectiveness of prompts depends on the user's proficiency in English, the output may vary significantly depending on the exact requirements of the user and how the LLM interprets these prompts. In addition, the prompts can unintentionally reinforce existing biases within the model's training data, potentially skewing the results. Therefore, prompt engineering must be approached carefully and methodically to minimise misinterpretation and maximise output relevance.

3.5. Autonomous Agents

The evolution of LLMs, with their ability to generate information and communicate in a manner that resembles human interaction, has led to the development of autonomous agents. The expectation is that these agents will effectively perform a wide range of tasks, taking advantage of the human-like capabilities inherent in LLMs [121]. These autonomous agents follow a four-stage architecture that includes profiling, memory, planning, and action. Profiling defines the agent's role, privileges, domain, and expertise [121]. Memory stores information on tasks and profile data relevant to the environment. Planning involves breaking down given tasks into subtasks and solving them individually. The action stage is the final phase where all decisions and subtasks are translated into actions executed by the agent. Zhang et al. [122] developed a framework designed to facilitate collaboration between GenAI agents and humans. This framework enables planning and communication for specific tasks, leveraging the capabilities of LLMs Similarly, AgentSims [123], ToolBench [124], GameGPT [125], ChatDev [126], Voyager [127], and RecMind [128] represent a diverse array of autonomous GenAI agents developed with distinct goals and objectives. Certainly, AutoGen stands out as a multiagent framework with the capability to autonomously perform tasks or collaborate with human feedback. This flexibility makes it a versatile tool for various applications [129].

In addition to the AutoGen framework, AgentLite [130], Camel [131] and CrewAI [132] are each similar LLM-based agent framework architectures. These platforms are distinguished by their support for task decomposition, multi-agent orchestration, and adaptable reasoning. In particular, AgentLite and CrewAI facilitate work delegation functionalities, increasing their utility in various operational contexts.

3.6. Retrieval-augmented Generation

The content generated by LLMs is highly dependent on the extensive text-based datasets on which they are trained. These datasets may contain vast amounts of information utilised by the LLMs. However, maintaining up-to-date knowledge within LLMs is challenging, as fine-tuning or retraining a model

is often extremely costly and resource-intensive. Retrieval-augmented Generation (RAG) is a technique designed to address this LLM knowledge gap by retrieving information from external sources and integrating it with the model's internal representations [133].

A major advantage of RAG is its ability to reduce the hallucination problem in LLMs, allowing them to generate more accurate and current information [134, 135]. The architecture of RAG consists of a knowledge base and a retriever model. The retriever model converts input prompts and content from the knowledge base to vectors. The user prompts are then appended with the most relevant content from the knowledge base, and this augmented prompt is sent to the base model to generate a more accurate response [135].

3.7. Limitations and Risks

As explored in the preceding sections, LLMs appear to possess a vast range of capabilities. However, it is crucial to acknowledge that they are not without limitations and risks. In multimodal LLMs, it is a common problem that they are over-reliant [136]. There are also potentially significant drawbacks associated with LLMs, including issues such as bias, explainability challenges, reasoning errors, logical errors, hallucinations, vulnerability to prompt injections, and spelling and grammar errors. These limitations underscore the importance of a cautious and critical approach when using LLMs in various applications. Furthermore, the literature shows limitations in LLMs, including statistical inconsistency, the absence of emotional attributes in linguistic responses, and challenges related to fact verification [137, 138]. These factors contribute to a comprehensive understanding of the constraints and potential shortcomings when working with LLMs. Thapa et al. [139] contend that while LLMs can indeed reduce the time and costs associated with annotation tasks, they are not completely supplanting human annotation. This is because they struggle with intricate linguistic constructions, such as idioms, irony, sarcasm, and metaphor, which can potentially impact the precision of annotations.

Similar limitations are associated with MLLMs, such as over-reliance on training data, sensitivity to word order in input prompts, and vulnerability to prompts containing additional knowledge [140]. There are several more concerns associated with LLMs, including restricted text input and output lengths, limited comprehension of syntax, ethical considerations with the generated information, constraints with multilingual capabilities, elevated costs associated with training and maintenance, inadequate understanding of human behaviours and limited ability to learn incrementally [141, 142].

Despite their considerable capabilities, LLMs are not without risks. Wiggins and Tejani [143], Lund and Wang [144], Rahman and Santacana [145] provide comprehensive overviews of risks linked to LLMs. These include the homogenisation of results, whereby defects or biases from the foundation model are inherited by all downstream models. There is also the risk of monopolistic control by foundation model owners, potentially concentrating decision-making power, resource access, and influence over model usage in a single entity. Ethical and legal concerns are intertwined with concerns about privacy and intellectual property. In addition, there are economic and environmental impacts that raise concerns about the potential displacement of human workers. Furthermore, inequity and misuse of LLMs, such as the creation of deepfakes and their application in criminal and unethical activities, pose additional challenges. Given that LLMs do not inherently prioritise the precision of information, Bender et al. [146] have highlighted the risk of generating social turbulence, especially when used on social media platforms. Furthermore, the use of LLMs is associated with significant costs, leading to a direct environmental impact due to their substantial energy consumption [147].

The risks associated with LLMs are predominantly emphasised within Information Communication and Technology (ICT) and cyberspace. Primary concerns include the disclosure of personal information, the generation of malicious text, and the creation of malicious code [148].

The Beyond the Imitation Game benchmark (BIG-bench), serves as an evaluation framework for LLMs. It encompasses 204 distinct language-related tasks. These span contextual and context-free question-answering, reading comprehension, logical reasoning, etc. [149]. It is acknowledged that the challenge of social biases and dependency on the English language persists in almost all LLMs.

When employing LLM-based agents, it is imperative to address challenges associated with LLM-based multi-agent frameworks as well. The handling of many defined agents may necessitate substantial computational resources and memory, thus mandating high-end computing infrastructure for seamless operations. Furthermore, the absence of a standardised comprehensive benchmarking system to evaluate the behaviour of such agents underscores the limitations inherent in the development of LLM-based multi-agent systems. These challenges underscore the need for further research and refinement in this domain to enhance the efficiency and effectiveness of LLM-based multi-agent frameworks [150].

Similarly to these risks, the Open Web Application Security Project (OWASP) has identified ten major risk factors related to LLMs. These risks include training data poisoning, prompt injection, denial of service, insecure output handling, supply chain vulnerabilities, sensitive information leakage, excessive agency, insecure plugins, overreliance, and model theft of data. Despite these threat factors, OWASP also stressed the need for regulatory bodies to supervise LLMs in various domains and recommended the implementation of risk management programmes that incorporate the checklist provided by OWASP[4].

The EU's Artificial Intelligence Act (AIA) proposes a framework for categorising AI applications based on their associated risk levels, with the primary aim of safeguarding human rights and maintaining ethical standards in AI deployment [151]. Within the AIA, AI applications are divided into categories such as "unacceptable risk", which includes

[4]https://owasp.org/www-project-top-10-for-large-language-model-applications

practices such as exploiting vulnerabilities and social-ranking techniques due to their potential for individual manipulation and impact on fundamental rights. These categories have relevance to DF, where the ethical application of LLMs must balance the advantages of automation with the imperative to address privacy concerns. Since DF involves sensitive data and influences legal outcomes, the use of LLMs must align with AIA's risk-based principles, ensuring transparency, accountability, and fair application. In future AI applications within DF, it is essential to implement appropriate measures to prevent biases and hallucinations to mitigate the risk of misuse of AI.

4. Large Language Models For Digital Forensics

Section 4 summarises existing work with LLMs in DF, the feasibility of employing them, and potential future directions. As discussed in Sections 2 and 3, despite the widespread use of LLMs in various fields to improve the efficiency and accuracy of tasks within specific domains, their application in the field of DF is still relatively new.

Conducting a thorough analysis of LLMs use in conjunction with the stages of the DF process model, as highlighted in Section 1.1.1, proves to be a valuable undertaking.

4.1. Incident Recognition Phase

In the initial phase of Casey's DF process model, which delineates the recognition of an incident, LLMs can serve as a valuable detection mechanism [152]. In cybercrime cases, the primary artefacts often involve data logs, data dumps and network dumps. Fine-tuning an LLM to monitor text-based logs and related files enables it to discern and identify potential or ongoing incidents within the environment. In network-related activities, anomaly detection plays a pivotal role in initiating an incident response. Various existing anomaly detection techniques are employed in systems for this purpose. Using their ability to identify patterns in a series of text data sets, LLMs exhibit potential as an Intrusion Detection System (IDS) within such systems [33, 153].

For instance, Kan et al. [154] introduces Mobile-LLaMA, an open source mobile network-specialised LLM, fine-tuned through instructional data to enhance their capabilities for network analysis tasks within 5G environments. Mobile-LLaMA supports three primary functions: IP routing analysis, packet analysis, and performance analysis.

4.2. Collection Phase

Although evidence collection or seizure traditionally involves physical tasks that require human interaction, LLMs can play a role in identifying and listing potential pieces of evidence at a crime scene. For example, in the examination of photographs or video records from a crime scene, an investigator can enlist the help of a MLLM such as LLaVa, GPT-4, or VisionLLM. These models are capable of processing information within the images and generating a text-based output, facilitating the interpretation and categorisation of visual data. Although this task may seem simple and within the capabilities of a human agent, the efficiency becomes particularly evident when dealing with a massive-scale investigation involving thousands of collected artefacts and photographs. Using an MLLM for initial processing can significantly save time, with human agents then focussing on the crucial task of verification and validation.

4.3. Preservation/Acquisition Phase

The preservation of evidence is centred on maintaining integrity. To achieve this, various tools such as EnCase and FTK Imager have been used, helping investigators streamline their work processes [155]. In the context of non-technical DF stakeholders being able to interrogate the evidence, it becomes feasible for a user to articulate their requirements/query in natural language. Subsequently, the LLM generates source code tailored to the specific need, executes the code on the data, and returns the result in consumable natural language.

LLMs specialised in code generation, such as StarCoder and Code LLaMA, can be fine-tuned and retrained for domain-specific tasks, including the preservation of disk evidence through customised code and script generation. These LLMs are capable of generating scripts or code snippets that create secure copies of disk images, metadata, and partition information, as well as automating cryptographic hashing and verification routines to maintain the evidence's integrity through checksums. Additionally, LLMs can assist in documenting preservation steps by generating logs and summaries for each stage of the disk preservation process, thereby supporting the chain of custody during acquisitions. However, despite these capabilities, human expertise remains essential for identifying and collecting potential sources of evidence during the preservation phase, as the application of LLMs in this stage is currently limited to lower-potential tasks.

In certain instances, the gathering of live data for forensic investigations becomes crucial, particularly data collected at the crime scene. For this purpose, investigators can use DFaaS platforms such as Hansken. Hansken possesses the ability to amalgamate custom extraction APIs for data extractions, and these APIs can be developed using code-generative LLMs [20]. This approach improves the adaptability and efficiency of the investigative process.

As stated in Section 3.5, the automation of code generation and unit testing can be facilitated by autonomous agents that use LLMs as their core. AutoGen, being an open source framework, provides the means to develop AI agents tailored for specific tasks. These AutoGen agents are not only customisable and conversational, but can also operate in various modes, employing combinations of LLMs, human inputs, and various tools [129].

Automated agents, particularly those developed within frameworks such as AutoGen, can be used in the preservation phase of investigations. These agents can be assigned specific tasks, such as acquiring disk images, generating disk hashes, retrieving disk metadata, and compiling acquisition reports. By defining precise roles and tasks for AI agents, it is possible

to streamline and standardise these preservation actions, improving the management of digital evidence [129, 156].

4.4. Examination Phase

This phase constitutes a crucial component of the investigation, playing a crucial role in elucidating the case through activities such as data recovery, collection, reduction, and classification. For each of these components, LLMs fine-tuned for scripting can significantly assist, especially at a larger scale. Within these components, tasks such as keyword search, file recovery, pattern matching, and fragment reassembly can be achieved with minimal technical knowledge using LLMs. LLMs can provide valuable assistance in these tasks by generating new codes, crafting regular expressions, generating passwords and/or password hash lists for decryption, and creating sample logs or files. LLMs can generate a set of instructions, queries, and Application Programming Interface (API) validations from natural language provided by a human. This opens up the possibility of integrating third-party tools like Scapy, tshark, John the Ripper, and others seamlessly into the investigative process, enhancing the toolkit available for DF investigations, and the ability to automate these processes enhances efficiency and effectiveness in the examination phase of the investigation.

The use of LAMs and VoT techniques in the examination phase can significantly enhance the efficiency of an investigation. Since LAMs and VoT specifically focus on task manipulation, investigators can offload some examination work to an LAM, which will then generate the final results from a series of subtasks. This approach can allow investigators to focus on higher-level analysis and decision making, thus streamlining the overall investigative process.

4.5. Analysis Phase

The analysis phase involves understanding the incident and obtaining a conclusive understanding based on the information collected during the examination phase. As also highlighted in Section 1.1.3, it has been demonstrated that LLMs are effective in case analysis [30]. The use of MLLMs, which possess the capability to interpret images, broadens the scope for analysing a crime case more comprehensively.

Using Gemini 1.5, Xu et al. [157] presented a tutorial on profiling a suspect's web history through an LLM. This case study demonstrates how an LLM can help identify potential motivations, personal interests, and psychological characteristics of the suspect. In conclusion, the authors suggest that such mechanisms could power AI-assisted tools, enabling law enforcement authorities to improve the identification of cybercriminals and malicious entities.

The Digital Forensic Cybercrime Language as a Service (DFClaaS) is an innovative system developed to address the complexities of text-based cybercrime [158]. Using natural NLP techniques, including LLMs, sentiment analysis, and lexicon analysis, DFClaaS aims to improve capabilities in incident reporting, analysis, and investigation. The primary objectives of DFClaaS include implementing microservices to address specific challenges, proposing an advanced system to improve incident handling, and providing valuable tools for DF investigators. Designed to serve individual users, organisations, and forensic professionals, DFClaaS is a versatile and effective resource in the ongoing fight against cybercrime.

LLMs can be specifically fine-tuned for the analysis of various data types, including log files, email contents, chat transcripts, call records, file metadata, hex dumps, memory dumps, and registry hives. Incorporating contents such as event logs, timestamps, and network traffic captures further enables the effective recreation of incidents by correlating each data set with the assistance of LLMs. In addition, MLLMs that are audio and video specific can assist in analysing content within these formats. This specialised capability can significantly reduce the time investigators spend analysing audio and video data during investigations.

The use of automated agents can effectively distribute the analysis workload. Moreover, leveraging Augmented Large Language Models (ALLMs) and RAG techniques can improve knowledge retrieval in real time, thus improving the accuracy of analysis and decision-making processes [159]. For example, integrating a source of intelligence with an RAG system can assist investigators in connecting the dots during a DF investigation.

Other than these applications, LLMs can increase productivity through enhanced information correlation during the analysis phase. Shafee et al. [160] suggest that LLMs hold significant potential for data correlation and cybersecurity applications. The referenced study evaluated the performance of various LLM-based chatbots, including ChatGPT, GPT4all, Dolly, Stanford Alpaca, Alpaca-LoRA, Falcon, and Vicuna, specifically for text classification and Named Entity Recognition (NER) tasks using OSINT data. The findings indicated that, although the commercial chatbot GPT-4 and the open-source GPT4all performed well in text classification, all tested LLM-based chatbots showed limitations and were less effective for cybersecurity entity recognition compared to specialised models. The study concludes that there remains room for improvement.

4.6. Reporting Phase

The quality and validity of the evidence, along with the thoroughness of the analysis, are encapsulated in the final report. The reporting phase holds significant weight, as the entire judgement may hinge on this crucial stage. Notably, DF is experiencing heightened scrutiny about the quality of the reports, emphasising the importance of precision and clarity in this phase [161]. As pointed out by Champod et al. [162], there is no standard framework for evaluating and reporting scientific findings to authorities and stakeholders. To provide assistance and alleviate scrutiny, incorporating LLMs for report creation is a viable solution.

While LLMs are inherently non-deterministic, adhering to investigation standards such as ISO/IEC 27043:2015 can establish robust processes around data integrity and evidence handling, even though these standards do not directly address the randomness or variability in LLM outputs. The ISO/IEC 27043:2015 standard provides guidelines for a consistent DF

investigation framework, focusing on maintaining procedural rigour rather than modifying model behaviour. Although it does not directly resolve issues of LLM determinism, it can serve as a protocol to ensure that procedures involving LLMs uphold investigative standards and maintain integrity throughout the process [163].

A preliminary feasibility study by Michelet and Breitinger [164] highlighted the potential of LLMs to assist in automating forensic report generation. These models can facilitate the creation of structured sections, such as methodologies, data analysis, and summaries, by generating coherent, case-specific insights from forensic data. Additionally, LLMs could automate the production of reports in alternative formats, such as HTML or LaTeX, which are frequently used for dynamic, web-based, or highly technical documentation.

4.7. Other Possibilities

Scanlon et al. [13] highlights that LLMs can play an important role in teaching scenarios. This involvement extends to activities such as storyboarding, creation of synthetic content, and synthetic character profiling.

Fine-tuned models could further enhance training by generating more complex, realistic case examples that challenge trainees with nuanced scenarios, providing a robust foundation for practical skills development. These models may also help translate technical findings into accessible language, facilitating communication of insights to non-specialists, such as judges or other stakeholders.

4.8. Discussion on Potential for LLMs in DF

To provide a comprehensive understanding of the potential use of LLMs, Table 3 clarifies the sample functionalities within the framework of the National Institute of Standards and Technology (NIST) Computer Forensics Tool Testing Program (CFTT), highlighting the usability of LLMs and example prompts. The CFTT project establishes overarching specifications to assess the capabilities of tools, a framework adopted by numerous prominent free and commercial tools [5].

The potential for having a positive impact on the typical phases of the investigation increases as one progresses through the typical order of the phases. For example, there is little improvement that can be made by an LLM or automated scripting during the identification or acquisition phases, but significant potential for aiding investigators in the reporting phase [164, 156] – these are first and foremost large *language* models. The low/medium/high potential outlined below evaluates each DF phase based on three key requirements: reliance on human expertise, physical versus digital evidence handling, and scope for automation, as explained below.

- **Low Potential for the Identification and Collection Phases**
  - High dependency on human involvement, expertise, and/or specialised knowledge.
  - Involves extensive handling of physical evidence.
  - Limited or no feasibility for automation.
- **Medium Potential for the Preservation Phase**
  - Requires some level of human involvement or expertise, but is not critical to the process.
  - Primarily deals with digital evidence, with minimal physical evidence handling.
  - Feasible for automation to a significant extent.
- **High Potential for the Examination, Analysis and Reporting Phases**
  - Human involvement is needed for expert verification of the conducted analysis.
  - Exclusively focused on digital evidence.
  - Many common tasks are suitable for significant support from LLMs.

With these possibilities, the scope for research in DF is vast. Future research could be extended to the generation of digital forensic reports, as well as the summarisation of these reports for non-technical users. This would save time, but can also lead to more consistent documentation compared to manual documentation. Given the capacity of LLMs to manage large textual datasets, exploring pattern recognition holds significant value, particularly for investigations requiring the detection of anomalies or outliers in chats, log events, or emails.

In addition, LLMs' ability to interpret the tone of messages or chats enables their application in the sentiment analysis of text-based evidence. There is also potential in fine-tuning LLMs for domain-specific tasks, such as network forensics, where LLMs could analyse log files and application data related to specific activities. Automating LLM-based DF tools could further enable investigators to generate customised reports using natural language queries.

A critical future research direction lies in the ethical and legal considerations of LLM-generated content. As the application of LLMs is still emerging, future studies should focus on developing appropriate benchmarks, standardisation protocols, and addressing legal aspects to ensure responsible use of this technology.

5. Challenges and Risks

This section discusses the challenges and risks of using LLMs in DF. Despite their promising potential, there are significant risk factors to consider. These risks can have severe consequences for DF if not adequately identified and considered in the DF process.

[5] https://www.nist.gov/itl/ssd/software-quality-group/computer-forensics-tool-testing-program-cftt

Table 3: DF functionalities by CFTT highlighting the usability of LLMs and example prompts

CFTT Functionality	DF Phase(s)	Usable LLMs/Agent Frameworks	Example Prompt
Cloud Data Extraction	Acquisition	LLaMA (Fine-Tuned), Code Llama, StarCoder, AutoGen or CrewAI	Retrieve all the data inside given S3 bucket by using given credentials
Deleted File Recovery Specs	Acquisition, Examination	LLaMA (Fine-Tuned), Code Llama, StarCoder, AutoGen or CrewAI	Find all the deleted files from the X disk image and recover them to Y location
Disk Imaging	Acquisition	LLaMA (Fine-Tuned), Code Llama, StarCoder, AutoGen or CrewAI	Get a full disk image from this computer and save it in Z location
Forensic File Carving	Acquisition, Examination	LLaMA (Fine-Tuned), Code Llama, StarCoder, AutoGen or CrewAI	Find all the deleted PDF files from X disk image
Forensic Media Preparation	Acquisition	LLaMA (Fine-Tuned), Code Llama, StarCoder, AutoGen or CrewAI	Prepare the given X device for new investigation
String Search	Examination	LLaMA (Fine-Tuned), Code Llama, StarCoder, AutoGen or CrewAI	Search all the files containing the email of mail@test.com
Mobile Devices	Examination	LLaMA (Fine-Tuned), LLaVa (Fine-Tuned), Code Llama, StarCoder, AutoGen or CrewAI	Find all the photos taken with a computer within the last 3 months
MS Windows Registry	Examination	LLaMA (Fine-Tuned), Code Llama, StarCoder, AutoGen or CrewAI	Find information about the users from a given Windows disk image
SQLite Databases	Examination	LLaMA (Fine-Tuned), Code Llama, StarCoder, AutoGen or CrewAI	Find the access time of user Y to application X using given SQLite databases

5.1. Challenges for LLMs in Digital Forensics

To optimise the results, the LLMs will likely need to be trained with specific forensic data (i.e., previous case data) to achieve the best results. Given the complexity and variation of the cases, it is questionable how good the training data are and whether there are sufficient data [165]. Any bias in training data can lead to skewed interpretations and unjust outcomes. This problem of bias can be mitigated by using diverse and representative datasets during the training phase, e.g., datasets that come from diverse sources, different case types and geographic regions. Furthermore, techniques such as data filtering, distribution reconstruction, rebalancing, regularisation, and prompting can be implemented to actively identify and correct biased patterns in the base data sets of the model and its outputs [166, 167]. These techniques involve adjusting model weights or incorporating fairness constraints during training to reduce the likelihood of biased predictions. Regular audit of the model's performance against fairness benchmarks is also crucial to ensure that it remains unbiased over time [168].

The experience level of investigators and the practical strategies employed in conducting investigations are challenging to replicate with LLMs. Initially, LLMs can excel in assisting with certain subtasks, such as parsing and data conversion, tasks in which output can be easily verified. However, when it comes to more interpretative or inferential tasks, LLMs' lack of inherent transparency introduces explainability challenges. Unlike deterministic software, whose logic can be easily traced, LLMs often act as black boxes, making it difficult to validate and understand the rationale behind their conclusions, particularly when these outputs extend beyond straightforward parsing into areas requiring judgement and reasoning. This underscores the importance of explainability in the application of LLMs to forensics, where understanding the basis of each result is crucial for accuracy and accountability [164].

Publicly hosted and maintained LLMs are generally unsuitable for casework due to the sensitivity of the evidence and information involved, which require strict privacy and security controls that cannot be reliably ensured on public platforms. Furthermore, managing the substantial infrastructure needed for LLM training and deployment is both energy and resource intensive, presenting a financial hurdle, especially for smaller forensic laboratories with limited budgets. Although methods like retrieval-augmented generation (RAG) or prompt engineering can reduce some of the computational load by tailoring responses with existing models, they still require powerful GPU resources to effectively run these models, adding to the cost and accessibility barriers. Centralised systems could be a viable option, but they require well-defined guidelines for data sharing and stringent security standards to safeguard sensitive information.

Although LLMs can serve as valuable tools to support forensic investigations, it should be recognised that they currently function best as an aid, not a substitute for human expertise [169]. There is a risk that people may place too much trust in the results generated by LLMs (over-reliance), which could lead to complacency and overlook the need for detailed human expert analysis and validation.

To mitigate the potential misuse of LLMs, many LLMs are subjected to censorship [170, 171]. Although this censorship may serve as a preventive measure against unethical use, it can pose challenges in the field of DF. For example, if an investigator seeks evidence related to 'drugs' or evidence of other illegal material, censorship of the LLM may restrict access to accu-

rate information related to the investigator's query. This limitation underscores the need for a nuanced approach to censorship in LLMs, balancing ethical considerations with the imperative of facilitating effective forensic investigations. In addition, the censorship of generic, publicly accessible LLMs further supports the argument for a discipline-specific DF LLM.

Finally, ethical and legal considerations must also be discussed. Determining accountability in cases where LLMs produce false information or are compromised by hacking. Clarifying responsibilities between developers, users, and regulators is crucial to establish a framework for accountability. If LLM generated DF results lead to incorrect information, the responsibility may lie with both the developers, for ensuring the model's accuracy, and the users, for appropriately interpreting and validating the results.

5.2. Risks of Integration

The integration of LLMs within the DF process comes with inherent risks, in addition to the general LLM limitations outlined in Section 3.7. In particular, in the examination, analysis, and reporting phases, the use of LLMs introduces the risk of producing inaccurate information, primarily due to the phenomenon of inheritance hallucinations associated with these models [164, 169]. Additionally, the biases and obscurities present in an inheritance model can significantly impact the performance of a DF-focused LLM – potentially leading to the unacceptable generation of biased or inaccurate information within the DF process.

Hallucinations in LLMs present a considerable risk, as they can produce information that appears credible but is incorrect. This can lead law enforcement authorities to form invalid assumptions and make flawed decisions based on unreliable results. Additionally, inherent biases in LLMs can influence investigative outcomes, which could affect the fairness and integrity of legal procedures. Data privacy concerns are also prominent, as sensitive information confidentiality may be compromised when using LLMs in DF processes. Together, these factors present substantial challenges to the reliable and ethical application of LLMs.

It is also crucial to acknowledge that DF LLMs, like any complex model, are susceptible to adversarial manipulation [172]. This vulnerability poses a substantial risk in the context of sensitive domains such as DF, where the integrity of the information obtained is paramount. Adversarial attacks can compromise the reliability of LLM-generated outputs, potentially influencing the outcomes of various phases within the DF process.

Indeed, despite incorporating human verification, outputs and reports generated by LLMs within DF applications may encounter challenges regarding acceptance within the legal systems of different countries. This highlights a significant usability risk associated with LLM-based DF applications, but one that can be carefully mitigated by limiting the technology's deployment as a human-in-the-loop investigative aid as opposed to directly feeding into any investigative/judicial decision-making processes.

Mitigating these changes and risks can be challenging, particularly in scenarios that involve adopting country-specific legal systems. However, there are potential strategies to address technical challenges such as hallucinations, censorship, and substantial infrastructure costs.

One solution to mitigate hallucination was suggested by Ji et al. [173], who proposed an interactive self-reflection method for generated knowledge and answers, an approach that has shown promise. Another method of reducing hallucinations is the use of RAG, which provides a larger knowledge base for LLMs to minimise unknown information [135]. Other methods such as knowledge graphs, bias detection mechanisms, active learning methods for LLMs, supervised fine-tuning strategies, hallucination mitigation frameworks, and new decoding strategies can also help mitigate hallucinations to some extent [174, 175].

Censorship issues can be addressed by fine-tuning the model with uncensored information, a technique already applied to the LLaMA and Mistral models, leading to the development of the Dolphin models. An example is the Dolphin-2.0-mistral-7b, which is an uncensored version of the Mistral 7B model [176].

The high infrastructure costs associated with these models can be mitigated by employing Data Forensics as a Service (DFaaS) platforms such as Hansken. With DFaaS, investigators only need to input queries related to their investigations using personal computers, while the platform manages the model maintenance and computational demands [20].

Despite these promising integration risks, the use of LLMs may face limitations in adaptability. The performance of an LLM is inherently tied to the dataset on which it was trained, which means that its ability to respond to new or emerging information is constrained. For example, if an LLM is tasked with identifying possible malware in a system, it may struggle to detect newer malware variants that were not part of its training data [31]. To mitigate such issues, LLMs need to be fine-tuned frequently, which poses its own challenges due to the significant computational power required for such operations.

6. Conclusion

The convergence of LLMs with an array of technologies represents exciting synergy. Although the utilisation of LLMs in the realm of DF is still in its nascent stages, there is evidence of their substantial potential to significantly increase the efficiency of investigations. The exploration of investments for LLMs across the entire DF process is considered, with the aim of improving the productivity and efficiency of investigations. Additionally, the integration of LLMs into current DF tools is posited to reduce user training times, as these models comprehend natural language input and provide output accordingly. In the dynamic landscape of LLM applications for DF, promising avenues for further exploration and advancement unfold.

Although the surge in LLM research is promising, it is crucial to balance enthusiasm with awareness of existing challenges. The propensity of LLMs to produce hallucinations highlights the need for human oversight in critical

decision-making processes, underscoring the irreplaceable value of human judgement, intuition, and expertise. A notable limitation is the language dependency issue, as most LLMs are predominantly trained on English data, reducing their effectiveness with non-English content. Furthermore, the deployment of LLMs in DF involves significant costs related to the infrastructure to process evidence. Questions also arise about the validation of task correctness and quality when automated by LLMs, as well as the legal and professional acceptance of results obtained with limited human intervention.

The trustworthiness of LLMs remains a debatable issue that requires careful attention. It is crucial to establish clear boundaries and measures to define LLM trustworthiness. Addressing this will be a key aspect in the field of DF, ensuring that LLMs can be trusted for accurate and secure analysis, with the explainability of their operations being paramount.

Integrating LLMs with automated agents offers a promising path to automating DF processes, potentially allowing multiple cases to be handled concurrently for more timely and precise outcomes. This integration could significantly streamline investigations. Future research should explore the role of LLMs and AI in the decision making of DF. It is essential to focus on validating LLM generated outputs to ensure their scope, accuracy, reliability, and trustworthiness in investigations. More studies comparing DF outcomes with and without LLM integration are critical, as they could highlight the benefits of LLMs and the controlled applicability of LLMs in DF and similar fields.

A future use case involves developing forensic-specific LLMs fine-tuned for automated examinations. These models could be optimised for script generation to support investigations where no existing tools are available, allowing forensic analysts to create customised solutions on demand. Integrating AI agents with these models could streamline evidence handling by allowing investigators to perform complex queries more intuitively, such as retrieving all messages from a specific date without the need to craft regular expressions.

In essence, while LLMs offer exciting prospects for the future of digital forensics, a balanced approach that integrates their strengths with human oversight is essential for harnessing their full potential. Inevitably, LLM-facilitated DF processes themselves will become the focus of future investigation.

References

[1] L. Ali, Cyber Crimes-A Constant Threat For The Business Sectors And Its Growth (A Study Of The Online Banking Sectors In GCC), Journal of Developing Areas 53 (2019) 253–265. URL: https://ideas.repec.org/a/jda/journl/vol.53year2019issue2pp.253-265.html.

[2] X. Du, M. Scanlon, Methodology for the Automated Metadata-Based Classification of Incriminating Digital Forensic Artefacts, in: Proceedings of the 14th International Conference on Availability, Reliability and Security, ARES '19, Association for Computing Machinery, New York, NY, USA, 2019. URL: https://doi.org/10.1145/3339252.3340517. doi:10.1145/3339252.3340517.

[3] P. P. Ray, ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope, Internet of Things and Cyber-Physical Systems 3 (2023) 121–154. URL: https://www.sciencedirect.com/science/article/pii/S266734522300024X. doi:https://doi.org/10.1016/j.iotcps.2023.04.003.

[4] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, et al., Language Models Are Few-Shot Learners, in: Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS'20, Curran Associates Inc., Red Hook, NY, USA, 2020. URL: https://papers.nips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

[5] D. Nozza, F. Bianchi, A. Lauscher, D. Hovy, et al., Measuring Harmful Sentence Completion in Language Models for LGBTQIA+ Individuals, in: Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion, Association for Computational Linguistics, 2022, pp. 26–34.

[6] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, et al., GPT-4 Technical Report, CoRR abs/2303.08774 (2023). doi:10.48550/ARXIV.2303.08774.

[7] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, et al., LLaMA: Open and Efficient Foundation Language Models abs/2302.13971 (2023). doi:10.48550/arXiv.2302.13971.

[8] DeepSeek-AI, A. Liu, B. Feng, B. Xue, B. Wang, et al., DeepSeek-V3 Technical Report, 2024. URL: https://arxiv.org/abs/2412.19437. arXiv:2412.19437.

[9] V. Baryamureeba, F. Tushabe, The enhanced digital investigation process model, Digital Investigation (2004).

[10] S. Mukherjee, S. Haque, Review Paper on Digital Forensics Practices: A Road Map for Building Digital Forensics Capability, Iconic Research and Engineering Journals 1 (2018) 96–99.

[11] X. Du, N.-A. Le-Khac, M. Scanlon, Evaluation of Digital Forensic Process Models with Respect to Digital Forensics as a Service, in: Proceedings of the 16th European Conference on Cyber Warfare and Security (ECCWS 2017), ACPI, Dublin, Ireland, 2017, pp. 573–581.

[12] E. Casey, Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet, Academic press, 2011.

[13] M. Scanlon, F. Breitinger, C. Hargreaves, J.-N. Hilgert, J. Sheppard, ChatGPT for digital forensic investigation: The good, the bad, and the unknown, Forensic Science International: Digital Investigation 46 (2023) 301609. URL: https://www.sciencedirect.com/science/article/pii/S266628172300121X. doi:https://doi.org/10.1016/j.fsidi.2023.301609.

[14] T. Wu, F. Breitinger, S. O'Shaughnessy, Digital forensic tools: Recent advances and enhancing the status quo, Forensic Science International: Digital Investigation 34 (2020) 300999. URL: https://doi.org/10.1016/j.fsidi.2020.300999.

[15] A. Al-Dhaqm, S. A. Razak, R. A. Ikuesan, V. R. Kebande, K. Siddique, A Review of Mobile Forensic Investigation Process Models, IEEE Access 8 (2020) 173359–173375. doi:10.1109/ACCESS.2020.3014615.

[16] R. Lutui, A multidisciplinary digital forensic investigation process model, Business Horizons 59 (2016) 593–604.

[17] S. S. Mir, U. Shoaib, M. S. Sarfraz, Analysis of digital forensic investigation models, Int. J. Comput. Sci. Inform. Secur 14 (2016).

[18] Y. Prayudi, A. Ashari, T. K. Priyambodo, The Framework to Support the Digital Evidence Handling: A Case Study of Procedures for the Management of Evidence in Indonesia, Journal of Cases on Information Technology (JCIT) 22 (2020) 51–71.

[19] R. van Baar, H. van Beek, E. van Eijk, Digital Forensics as a Service: A game changer, Digital Investigation 11 (2014) S54–S62. URL: https://www.sciencedirect.com/science/article/pii/S1742287614000127. doi:https://doi.org/10.1016/j.diin.2014.03.007, proceedings of the First Annual DFRWS Europe.

[20] H. van Beek, E. van Eijk, R. van Baar, M. Ugen, J. Bodde, A. Siemelink, Digital Forensics as a Service: Game on, Digital Investigation 15 (2015) 20–38. URL: https://www.sciencedirect.com/science/article/pii/S1742287615000857. doi:https://doi.org/10.1016/j.diin.2015.07.004, special Issue: Big Data and Intelligent Data Analysis.

[21] H. M. van Beek, J. van den Bos, A. Boztas, E. Van Eijk, R. Schramp, M. Ugen, Digital forensics as a service: Stepping up the game, Forensic Science International: Digital Investigation 35 (2020) 301021.

[22] H. Dubey, S. Bhatt, L. Negi, Digital Forensics Techniques and Trends: A Review, The International Arab Journal of Information Technology (IAJIT) 20 (2023) 644–654. doi:10.34028/iajit/20/4/11.

[23] E. Kalaimannan, J. N. Gupta, S.-M. Yoo, Maximizing Investigation Ef-

fectiveness in Digital Forensic Cases, in: 2013 International Conference on Social Computing, 2013, pp. 618–623. doi:10.1109/SocialCom.2013.93.

[24] C. S. Koper, C. Lum, J. J. Willis, Optimizing the Use of Technology in Policing: Results and Implications from a Multi-Site Study of the Social, Organizational, and Behavioural Aspects of Implementing Police Technologies, Policing: A Journal of Policy and Practice 8 (2014) 212–221. URL: https://doi.org/10.1093/police/pau015. doi:10.1093/police/pau015.

[25] E. A. Vincze, Challenges in digital forensics, Police Practice and Research 17 (2016) 183–194. URL: https://doi.org/10.1080/15614263.2015.1128163. doi:10.1080/15614263.2015.1128163.

[26] G. Michelet, F. Breitinger, G. Horsman, Automation for digital forensics: Towards a definition for the community, Forensic Science International 349 (2023) 111769. URL: https://www.sciencedirect.com/science/article/pii/S0379073823002190. doi:https://doi.org/10.1016/j.forsciint.2023.111769.

[27] The impact of automation and artificial intelligence on digital forensics, WIREs Forensic Science 3 (2021) e1418. URL: https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wfs2.1418. doi:https://doi.org/10.1002/wfs2.1418.

[28] A. Wickramasekara, A. Densmore, F. Breitinger, H. Studiawan, M. Scanlon, AutoDFBench: A Framework for AI Generated Digital Forensic Code and Tool Testing and Evaluation, in: Digital Forensics Doctoral Symposium, DFDS 2025, Association for Computing Machinery, New York, NY, USA, 2025. URL: https://doi.org/10.1145/3712716.3712718. doi:10.1145/3712716.3712718.

[29] S. Silalahi, T. Ahmad, H. Studiawan, Transformer-based Sentiment Analysis for Anomaly Detection on Drone Forensic Timeline, in: 2023 11th International Symposium on Digital Forensics and Security (ISDFS), 2023, pp. 1–6. doi:10.1109/ISDFS58141.2023.10131749.

[30] H. Henseler, H. van Beek, ChatGPT as a Copilot for Investigating Digital Evidence, in: J. G. Conrad, D. W. L. Jr., J. R. Baron, H. Henseler, P. Bhattacharya, A. Nielsen, J. K. Vinjumur, J. Pickens, A. Jones (Eds.), Proceedings of the Third International Workshop on Artificial Intelligence and Intelligent Assistance for Legal Professionals in the Digital Workplace (LegalAIIA 2023) co-located with the 19th International Conference on Artificial Intelligence and Law (ICAIL 2023), Braga, Portugal, June 19, 2023, volume 3423 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 58–69. URL: https://ceur-ws.org/Vol-3423/paper6.pdf.

[31] Z. Yu, M. Wen, X. Guo, H. Jin, Maltracker: A Fine-Grained NPM Malware Tracker Copiloted by LLM-Enhanced Dataset, in: Proceedings of the 33rd ACM SIGSOFT International Symposium on Software Testing and Analysis, ISSTA 2024, Association for Computing Machinery, New York, NY, USA, 2024, p. 1759–1771. URL: https://doi.org/10.1145/3650212.3680397. doi:10.1145/3650212.3680397.

[32] E. Karlsen, X. Luo, N. Zincir-Heywood, M. Heywood, Benchmarking Large Language Models for Log Analysis, Security, and Interpretation, Journal of Network and Systems Management 32 (2024) 59. URL: https://doi.org/10.1007/s10922-024-09831-x. doi:10.1007/s10922-024-09831-x.

[33] O. G. Lira, A. Marroquin, M. A. To, "Harnessing the Advanced Capabilities of LLM for Adaptive Intrusion Detection Systems", in: L. Barolli (Ed.), Advanced Information Networking and Applications, Springer Nature Switzerland, Cham, 2024, pp. 453–464.

[34] G. Lu, X. Ju, X. Chen, W. Pei, Z. Cai, GRACE: Empowering LLM-based software vulnerability detection with graph structure and in-context learning, Journal of Systems and Software 212 (2024) 112031. URL: https://www.sciencedirect.com/science/article/pii/S0164121224000748. doi:https://doi.org/10.1016/j.jss.2024.112031.

[35] R. Dale, GPT-3: What's it good for?, Natural Language Engineering 27 (2021) 113–118.

[36] D. E. O'Leary, An analysis of three chatbots: BlenderBot, ChatGPT and LaMDA, Intelligent Systems in Accounting, Finance and Management 30 (2023) 41–54. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/isaf.1531. doi:https://doi.org/10.1002/isaf.1531.

[37] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, et al., PaLM: Scaling Language Modeling with Pathways, Journal of Machine Learning Research 24 (2023) 1–113.

[38] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, in: J. Burstein, C. Doran, T. Solorio (Eds.), Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), Association for Computational Linguistics, 2019, pp. 4171–4186. URL: https://doi.org/10.18653/v1/n19-1423. doi:10.18653/V1/N19-1423.

[39] A. Reshamwala, D. Mishra, P. Pawar, Review on natural language processing, IRACST Engineering Science and Technology: An International Journal (ESTIJ) 3 (2013) 113–116.

[40] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444. URL: https://doi.org/10.1038/nature14539.

[41] S. Dong, P. Wang, K. Abbas, A survey on deep learning and its applications, Computer Science Review 40 (2021) 100379. URL: https://www.sciencedirect.com/science/article/pii/S1574013721000198. doi:https://doi.org/10.1016/j.cosrev.2021.100379.

[42] J. Yang, H. Jin, R. Tang, X. Han, Q. Feng, H. Jiang, S. Zhong, B. Yin, X. Hu, Harnessing the Power of LLMs in Practice: A Survey on ChatGPT and Beyond, ACM Trans. Knowl. Discov. Data 18 (2024). URL: https://doi.org/10.1145/3649506. doi:10.1145/3649506.

[43] Y. Shen, L. Heacock, J. Elias, K. D. Hentel, B. Reig, G. Shih, L. Moy, ChatGPT and Other Large Language Models Are Double-edged Swords, Radiology 307 (2023) e230163. URL: https://doi.org/10.1148/radiol.230163. doi:10.1148/radiol.230163, pMID: 36700838.

[44] S. Lai, K. Liu, S. He, J. Zhao, How to generate a good word embedding, IEEE Intelligent Systems 31 (2016) 5–14.

[45] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient Estimation of Word Representations in Vector Space, in: Y. Bengio, Y. LeCun (Eds.), 1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings, 2013.

[46] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is All you Need, in: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems, volume 30, Curran Associates, Inc., 2017.

[47] G. Ke, D. He, T. Liu, Rethinking Positional Encoding in Language Pretraining, in: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021, 2021.

[48] R. Al-Rfou, D. Choe, N. Constant, M. Guo, L. Jones, Character-level language modeling with deeper self-attention, in: Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI'19/IAAI'19/EAAI'19, AAAI Press, 2019. URL: https://doi.org/10.1609/aaai.v33i01.33013159. doi:10.1609/aaai.v33i01.33013159.

[49] F. A. Acheampong, H. Nunoo-Mensah, W. Chen, Transformer models for text-based emotion detection: a review of BERT-based approaches, Artificial Intelligence Review 54 (2021) 1–41. doi:https://doi.org/10.1007/s10462-021-09958-2.

[50] A. M. Bran, S. Cox, O. Schilter, C. Baldassari, A. D. White, P. Schwaller, Augmenting large language models with chemistry tools, Nature Machine Intelligence 6 (2024) 525–535. URL: https://doi.org/10.1038/s42256-024-00832-8. doi:10.1038/s42256-024-00832-8.

[51] M.-L. Tsai, C. W. Ong, C.-L. Chen, Exploring the use of large language models (LLMs) in chemical engineering education: Building core course problem models with Chat-GPT, Education for Chemical Engineers 44 (2023) 71–95. URL: https://www.sciencedirect.com/science/article/pii/S1749772823000180. doi:https://doi.org/10.1016/j.ece.2023.05.001.

[52] X. Hou, Y. Zhao, Y. Liu, Z. Yang, K. Wang, L. Li, X. Luo, D. Lo, J. Grundy, H. Wang, Large Language Models for Software Engineering: A Systematic Literature Review, ACM Trans. Softw. Eng. Methodol. (2024). URL: https://doi.org/10.1145/3695988. doi:10.1145/

3695988, just Accepted.

[53] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, W. Ye, Y. Zhang, Y. Chang, P. S. Yu, Q. Yang, X. Xie, A Survey on Evaluation of Large Language Models, ACM Trans. Intell. Syst. Technol. (2024). URL: https://doi.org/10.1145/3641289. doi:10.1145/3641289.

[54] M. Vidgof, S. Bachhofner, J. Mendling, Large Language Models for Business Process Management: Opportunities and Challenges, in: C. Di Francescomarino, A. Burattin, C. Janiesch, S. Sadiq (Eds.), Business Process Management Forum, Springer Nature Switzerland, Cham, 2023, pp. 107–123.

[55] I. Carvalho, S. Ivanov, ChatGPT for tourism: applications, benefits and risks, Tourism Review (2023). URL: https://doi.org/10.1108/TR-02-2023-0088. doi:10.1108/TR-02-2023-0088.

[56] S. Moore, R. Tong, A. Singh, Z. Liu, X. Hu, et al., Empowering Education with LLMs: The Next-Gen Interface and Content Generation, in: International Conference on Artificial Intelligence in Education, Springer, 2023, pp. 32–37.

[57] J. Liu, C. S. Xia, Y. Wang, L. Zhang, Is Your Code Generated by ChatGPT Really Correct? Rigorous Evaluation of Large Language Models for Code Generation, in: A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine (Eds.), Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023, 2023. URL: http://papers.nips.cc/paper_files/paper/2023/hash/43e9d647ccd3e4b7b5baab53f0368686-Abstract-Conference.html.

[58] H. Li, J. Zhang, H. Liu, J. Fan, X. Zhang, J. Zhu, R. Wei, H. Pan, C. Li, H. Chen, CodeS: Towards Building Open-source Language Models for Text-to-SQL, Proc. ACM Manag. Data 2 (2024). URL: https://doi.org/10.1145/3654930. doi:10.1145/3654930.

[59] F. Eggmann, R. Weiger, N. U. Zitzmann, M. B. Blatz, Implications of large language models such as ChatGPT for dental medicine, Journal of Esthetic and Restorative Dentistry (2023). URL: https://doi.org/10.1111/jerd.13046.

[60] A. J. Thirunavukarasu, D. S. J. Ting, K. Elangovan, L. Gutierrez, T. F. Tan, D. S. W. Ting, Large language models in medicine, Nature Medicine 29 (2023) 1930–1940.

[61] M. Karabacak, K. Margetis, Embracing large language models for medical applications: Opportunities and challenges, Cureus 15 (2023). URL: https://doi.org/10.7759/cureus.39305.

[62] E. Bonner, R. Lege, E. Frazier, Large Language Model-Based Artificial Intelligence in the Language Classroom: Practical Ideas for Teaching, Teaching English with Technology 23 (2023). URL: https://doi.org/10.56297/bkam1691/wieo1749. doi:10.56297/bkam1691/wieo1749.

[63] A. Caines, L. Benedetto, S. Taslimipoor, C. Davis, et al., On the Application of Large Language Models for Language Teaching and Assessment Technology, in: International Conference on Artificial Intelligence in Education (AIED), volume 3487 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 173–197.

[64] Z. Liu, Y. Zhou, Y. Zhu, J. Lian, C. Li, Z. Dou, D. Lian, J.-Y. Nie, Information Retrieval Meets Large Language Models, in: Companion Proceedings of the ACM Web Conference 2024, WWW '24, Association for Computing Machinery, New York, NY, USA, 2024, p. 1586–1589. URL: https://doi.org/10.1145/3589335.3641299. doi:10.1145/3589335.3641299.

[65] M. Bakker, M. Chadwick, H. Sheahan, M. Tessler, L. Campbell-Gillingham, J. Balaguer, N. McAleese, A. Glaese, J. Aslanides, M. Botvinick, C. Summerfield, Fine-tuning language models to find agreement among humans with diverse preferences, in: S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, A. Oh (Eds.), Advances in Neural Information Processing Systems, volume 35, Curran Associates, Inc., 2022, pp. 38176–38189. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/f978c8f3b5f399cae464e85f72e28503-Paper-Conference.pdf.

[66] W. Tan, C. Ding, J. Jiang, F. Wang, Y. Zhan, D. Tao, Harnessing the Power of MLLMs for Transferable Text-to-Image Person ReID, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024, pp. 17127–17137.

[67] W. Wu, S. Mao, Y. Zhang, Y. Xia, L. Dong, L. Cui, F. Wei, Mind's eye of llms: Visualization-of-thought elicits spatial reasoning in large language models, 2024. URL: https://api.semanticscholar.org/CorpusID:268889526.

[68] H. Zhao, H. Chen, F. Yang, N. Liu, H. Deng, H. Cai, S. Wang, D. Yin, M. Du, Explainability for Large Language Models: A Survey, ACM Trans. Intell. Syst. Technol. 15 (2024). URL: https://doi.org/10.1145/3639372. doi:10.1145/3639372.

[69] U. Alon, R. Sadaka, O. Levy, E. Yahav, Structural language models of code, in: Proceedings of the 37th International Conference on Machine Learning, ICML'20, JMLR.org, 2020. URL: https://dl.acm.org/doi/10.5555/3524938.3524962. doi:10.5555/3524938.3524962.

[70] F. F. Xu, U. Alon, G. Neubig, V. J. Hellendoorn, A Systematic Evaluation of Large Language Models of Code, in: Proceedings of the 6th ACM SIGPLAN International Symposium on Machine Programming, MAPS 2022, Association for Computing Machinery, New York, NY, USA, 2022, p. 1–10. URL: https://doi.org/10.1145/3520312.3534862. doi:10.1145/3520312.3534862.

[71] M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. de Oliveira Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, A. Ray, R. Puri, G. Krueger, M. Petrov, H. Khlaaf, G. Sastry, P. Mishkin, B. Chan, S. Gray, N. Ryder, M. Pavlov, A. Power, L. Kaiser, M. Bavarian, C. Winter, P. Tillet, F. P. Such, D. Cummings, M. Plappert, F. Chantzis, E. Barnes, A. Herbert-Voss, W. H. Guss, A. Nichol, A. Paino, N. Tezak, J. Tang, I. Babuschkin, S. Balaji, S. Jain, W. Saunders, C. Hesse, A. N. Carr, J. Leike, J. Achiam, V. Misra, E. Morikawa, A. Radford, M. Knight, M. Brundage, M. Murati, K. Mayer, P. Welinder, B. McGrew, D. Amodei, S. McCandlish, I. Sutskever, W. Zaremba, Evaluating Large Language Models Trained on Code, CoRR abs/2107.03374 (2021). doi:10.48550/arXiv.2107.03374.

[72] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, et al., Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, volume 35, 2022, pp. 24824–24837.

[73] B. Rozière, J. Gehring, F. Gloeckle, S. Sootla, I. Gat, X. E. Tan, Y. Adi, J. Liu, T. Remez, J. Rapin, A. Kozhevnikov, I. Evtimov, J. Bitton, M. Bhatt, C. Canton-Ferrer, A. Grattafiori, W. Xiong, A. Défossez, J. Copet, F. Azhar, H. Touvron, L. Martin, N. Usunier, T. Scialom, G. Synnaeve, Code Llama: Open Foundation Models for Code, CoRR (2023). doi:10.48550/arXiv.2308.12950.

[74] E. Nijkamp, B. Pang, H. Hayashi, L. Tu, H. Wang, et al., CodeGen: An Open Large Language Model for Code with Multi-Turn Program Synthesis, in: The Eleventh International Conference on Learning Representations, 2022.

[75] R. Li, L. B. Allal, Y. Zi, N. Muennighoff, D. Kocetkov, et al., StarCoder: May the Source Be with You!, Transactions on Machine Learning Research (2023).

[76] H. Yu, B. Shen, D. Ran, J. Zhang, Q. Zhang, Y. Ma, G. Liang, Y. Li, Q. Wang, T. Xie, CoderEval: A Benchmark of Pragmatic Code Generation with Generative Pre-trained Models, in: Proceedings of the IEEE/ACM 46th International Conference on Software Engineering, ICSE '24, Association for Computing Machinery, New York, NY, USA, 2024. URL: https://doi.org/10.1145/3597503.3623316. doi:10.1145/3597503.3623316.

[77] Z. Luo, C. Xu, P. Zhao, Q. Sun, X. Geng, W. Hu, C. Tao, J. Ma, Q. Lin, D. Jiang, WizardCoder: Empowering Code Large Language Models with Evol-Instruct, in: The Twelfth International Conference on Learning Representations, 2024.

[78] D. Fried, A. Aghajanyan, J. Lin, S. Wang, E. Wallace, F. Shi, R. Zhong, S. Yih, L. Zettlemoyer, M. Lewis, InCoder: A Generative Model for Code Infilling and Synthesis, in: The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023, 2023.

[79] D. Zhou, N. Schärli, L. Hou, J. Wei, N. Scales, X. Wang, D. Schuurmans, C. Cui, O. Bousquet, Q. V. Le, et al., Least-to-Most Prompting Enables Complex Reasoning in Large Language Models, in: The Eleventh International Conference on Learning Representations, 2022.

[80] Y. Wang, H. Le, A. Gotmare, N. Bui, J. Li, S. Hoi, CodeT5+: Open Code Large Language Models for Code Understanding and Generation (2023) 1069–1088. doi:10.18653/v1/2023.emnlp-main.68.

[81] Y. Wang, W. Wang, S. Joty, S. C. Hoi, CodeT5: Identifier-

aware Unified Pre-trained Encoder-Decoder Models for Code Understanding and Generation, in: M.-F. Moens, X. Huang, L. Specia, S. W.-t. Yih (Eds.), Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 2021, pp. 8696–8708. URL: https://aclanthology.org/2021.emnlp-main.685/. doi:10.18653/v1/2021.emnlp-main.685.

[82] B. Chen, F. Zhang, A. Nguyen, D. Zan, Z. Lin, J.-G. Lou, W. Chen, CodeT: Code Generation with Generated Tests, in: The Eleventh International Conference on Learning Representations, 2023, pp. 1069,1088. URL: https://doi.org/10.18653/v1/2023.emnlp-main.68.

[83] N. Shinn, F. Cassano, A. Gopinath, K. R. Narasimhan, S. Yao, Reflexion: Language Agents with Verbal Reinforcement Learning, in: Thirty-seventh Conference on Neural Information Processing Systems, 2023.

[84] Q. Zheng, X. Xia, X. Zou, Y. Dong, S. Wang, et al., CodeGeeX: A Pre-Trained Model for Code Generation with Multilingual Benchmarking on HumanEval-X (2023) 5673–5684. URL: https://doi.org/10.1145/3580305.3599790. doi:10.1145/3580305.3599790.

[85] Y. Li, D. Choi, J. Chung, N. Kushman, J. Schrittwieser, R. Leblond, T. Eccles, J. Keeling, F. Gimeno, A. D. Lago, T. Hubert, P. Choy, C. de Masson d'Autume, I. Babuschkin, X. Chen, P.-S. Huang, J. Welbl, S. Gowal, A. Cherepanov, J. Molloy, D. J. Mankowitz, E. S. Robson, P. Kohli, N. de Freitas, K. Kavukcuoglu, O. Vinyals, Competition-level code generation with AlphaCode, Science 378 (2022) 1092–1097. URL: https://www.science.org/doi/abs/10.1126/science.abq1158. doi:10.1126/science.abq1158.

[86] L. B. Allal, R. Li, D. Kocetkov, C. Mou, C. Akiki, et al., SantaCoder: don't reach for the stars!, Deep Learning for Code (DL4C) Workshop, 2023. URL: https://par.nsf.gov/biblio/10416454.

[87] P. Di, J. Li, H. Yu, W. Jiang, W. Cai, Y. Cao, C. Chen, D. Chen, H. Chen, L. Chen, G. Fan, J. Gong, Z. Gong, W. Hu, T. Guo, Z. Lei, T. Li, Z. Li, M. Liang, C. Liao, B. Liu, J. Liu, Z. Liu, S. Lu, M. Shen, G. Wang, H. Wang, Z. Wang, Z. Xu, J. Yang, Q. Ye, G. Zhang, Y. Zhang, Z. Zhao, X. Zheng, H. Zhou, L. Zhu, X. Zhu, CodeFuse-13B: A Pretrained Multilingual Code Large Language Model, in: Proceedings of the 46th International Conference on Software Engineering: Software Engineering in Practice, ICSE-SEIP '24, Association for Computing Machinery, New York, NY, USA, 2024, p. 418–429. URL: https://doi.org/10.1145/3639477.3639719. doi:10.1145/3639477.3639719.

[88] K. S. Kalyan, A survey of GPT-3 family large language models including ChatGPT and GPT-4, Natural Language Processing Journal 6 (2024) 100048. URL: https://www.sciencedirect.com/science/article/pii/S2949719123000456. doi:https://doi.org/10.1016/j.nlp.2023.100048.

[89] Z. Yu, X. Zhang, N. Shang, Y. Huang, C. Xu, Y. Zhao, W. Hu, Q. Yin, WaveCoder: Widespread And Versatile Enhancement For Code Large Language Models By Instruction Tuning, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 5140–5153. doi:10.18653/v1/2024.acl-long.280.

[90] Z. Yu, Y. Zhao, A. Cohan, X.-P. Zhang, HumanEval Pro and MBPP Pro: Evaluating Large Language Models on Self-invoking Code Generation, 2024. URL: https://arxiv.org/abs/2412.21199. arXiv:2412.21199.

[91] M. G. Rizzo, N. Cai, D. Constantinescu, The performance of ChatGPT on orthopaedic in-service training exams: A comparative study of the GPT-3.5 turbo and GPT-4 models in orthopaedic education, Journal of Orthopaedics 50 (2024) 70–75. URL: https://www.sciencedirect.com/science/article/pii/S0972978X2300332X. doi:https://doi.org/10.1016/j.jor.2023.11.056.

[92] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, et al., Learning Transferable Visual Models From Natural Language Supervision, in: Proceedings of the 38th International Conference on Machine Learning, volume 139 of *Proceedings of Machine Learning Research*, PMLR, 2021, pp. 8748–8763.

[93] Y. Tewel, Y. Shalev, I. Schwartz, L. Wolf, ZeroCap: Zero-Shot Image-to-Text Generation for Visual-Semantic Arithmetic, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 17918–17928.

[94] J. Zhang, J. Huang, S. Jin, S. Lu, Vision-Language Models for Vision Tasks: A Survey, IEEE Transactions on Pattern Analysis and Machine Intelligence 46 (2024) 5625–5644. doi:10.1109/TPAMI.2024.3369699.

[95] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, et al., Zero-Shot Text-to-Image Generation, in: Proceedings of the 38th International Conference on Machine Learning, volume 139 of *Proceedings of Machine Learning Research*, PMLR, 2021, pp. 8821–8831.

[96] J. Y. Koh, D. Fried, R. Salakhutdinov, Generating Images with Multimodal Language Models, in: A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine (Eds.), Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023, 2023. URL: http://papers.nips.cc/paper_files/paper/2023/hash/43a69d143273bd8215578bde887bb552-Abstract-Conference.html.

[97] W. Wang, Z. Chen, X. Chen, J. Wu, X. Zhu, G. Zeng, P. Luo, T. Lu, J. Zhou, Y. Qiao, J. Dai, VisionLLM: Large Language Model is also an Open-Ended Decoder for Vision-Centric Tasks, in: A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine (Eds.), Advances in Neural Information Processing Systems, volume 36, Curran Associates, Inc., 2023, pp. 61501–61513. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/c1f7b1ed763e9c75e4db74b49b76db5f-Paper-Conference.pdf.

[98] H. Liu, C. Li, Q. Wu, Y. J. Lee, Visual Instruction Tuning, in: A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine (Eds.), Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023, 2023. URL: http://papers.nips.cc/paper_files/paper/2023/hash/6dcf277ea32ce3288914faf369fe6de0-Abstract-Conference.html.

[99] H. Liu, C. Li, Y. Li, Y. J. Lee, Improved baselines with visual instruction tuning, in: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024, pp. 26286–26296. doi:10.1109/CVPR52733.2024.02484.

[100] D. Zhu, J. Chen, X. Shen, X. Li, M. Elhoseiny, MiniGPT-4: Enhancing Vision-Language Understanding with Advanced Large Language Models, in: The Twelfth International Conference on Learning Representations, 2024.

[101] Q. Guo, S. De Mello, H. Yin, W. Byeon, K. C. Cheung, Y. Yu, P. Luo, S. Liu, RegionGPT: Towards Region Understanding Vision Language Model, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024, pp. 13796–13806.

[102] A. Thapa, R. Patil, ChatGPT based ChatBot Application, in: IEEE SoutheastCon, 2024, pp. 157–164. doi:10.1109/SoutheastCon52093.2024.10500264.

[103] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, et al., Flamingo: a Visual Language Model for Few-Shot Learning, in: Advances in Neural Information Processing Systems, volume 35, Curran Associates, Inc., 2022, pp. 23716–23736. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/960a172bc7fbf0177ccccbb411a7d800-Paper-Conference.pdf.

[104] J. Li, D. Li, S. Savarese, S. Hoi, BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models, in: Proceedings of the 40th International Conference on Machine Learning, ICML'23, JMLR.org, 2023, pp. 19730–19742.

[105] J. Wu, W. Gan, Z. Chen, S. Wan, P. S. Yu, Multimodal Large Language Models: A Survey, in: 2023 IEEE International Conference on Big Data (BigData), 2023, pp. 2247–2256. doi:10.1109/BigData59044.2023.10386743.

[106] Y. Zhao, I. Misra, P. Krähenbühl, R. Girdhar, Learning Video Representations From Large Language Models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 6586–6597. URL: https://doi.org/10.1109/CVPR52729.2023.00637.

[107] S.-M. Park, Y.-G. Kim, Visual language integration: A survey and open challenges, Computer Science Review 48 (2023) 100548. URL: https://www.sciencedirect.com/science/article/pii/S1574013723000151. doi:https:

//doi.org/10.1016/j.cosrev.2023.100548.

[108] Z. Wang, M. Li, R. Xu, L. Zhou, J. Lei, et al., Language Models with Image Descriptors are Strong Few-Shot Video-Language Learners, in: Advances in Neural Information Processing Systems, volume 35, Curran Associates, Inc., 2022, pp. 8483–8497. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/381ceeae4a1feb1abc59c773f7e61839-Paper-Conference.pdf.

[109] K. Ma, X. Zang, Z. Feng, H. Fang, C. Ban, Y. Wei, Z. He, Y. Li, H. Sun, LLaViLo: Boosting Video Moment Retrieval via Adapter-Based Multimodal Modeling, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, 2023, pp. 2798–2803.

[110] R. Kiros, R. Salakhutdinov, R. Zemel, Multimodal Neural Language Models, in: Proceedings of the 31st International Conference on Machine Learning, volume 32 of *Proceedings of Machine Learning Research*, PMLR, Bejing, China, 2014, pp. 595–603. URL: https://proceedings.mlr.press/v32/kiros14.html.

[111] S. Uppal, S. Bhagat, D. Hazarika, N. Majumder, S. Poria, et al., Multimodal research in vision and language: A review of current and emerging trends, Information Fusion 77 (2022) 149–171. URL: https://www.sciencedirect.com/science/article/pii/S1566253521001512. doi:https://doi.org/10.1016/j.inffus.2021.07.009.

[112] Y. Zhang, S. Sun, M. Galley, Y.-C. Chen, C. Brockett, et al., DIALOGPT : Large-Scale Generative Pre-training for Conversational Response Generation, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Association for Computational Linguistics, Online, 2020, pp. 270–278. URL: https://doi.org/10.18653/v1/2020.acl-demos.30. doi:10.18653/v1/2020.acl-demos.30.

[113] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer, QLoRA: Efficient Finetuning of Quantized LLMs (2023). URL: http://papers.nips.cc/paper_files/paper/2023/hash/1feb87871436031bdc0f2beaa62a049b-Abstract-Conference.html.

[114] G. Penedo, Q. Malartic, D. Hesslow, R. Cojocaru, H. Alobeidli, A. Cappelli, B. Pannier, E. Almazrouei, J. Launay, The RefinedWeb dataset for falcon LLM: outperforming curated corpora with web data only, in: Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23, Curran Associates Inc., Red Hook, NY, USA, 2023. URL: https://dl.acm.org/doi/10.5555/3666122.3669586.

[115] J. Ou, J. Lu, C. Liu, Y. Tang, F. Zhang, D. Zhang, K. Gai, DialogBench: Evaluating LLMs as Human-like Dialogue Systems, in: K. Duh, H. Gomez, S. Bethard (Eds.), Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 6137–6170. doi:10.18653/v1/2024.naacl-long.341.

[116] G. Marvin, N. Hellen, D. Jjingo, J. Nakatumba-Nabende, Prompt Engineering in Large Language Models, in: I. J. Jacob, S. Piramuthu, P. Falkowski-Gilski (Eds.), Data Intelligence and Cognitive Informatics, Springer Nature Singapore, Singapore, 2024, pp. 387–402.

[117] Y. Zhou, A. I. Muresanu, Z. Han, K. Paster, S. Pitis, H. Chan, J. Ba, Large Language Models are Human-Level Prompt Engineers, in: The Eleventh International Conference on Learning Representations, 2023.

[118] M. B, Prompt Engineering as an Important Emerging Skill for Medical Professionals: Tutorial, J Med Internet Res 2023;25:e50638 (2022). URL: https://www.jmir.org/2023/1/e50638. doi:10.2196/50638.

[119] M. P. Polak, D. Morgan, Extracting accurate materials data from research papers with conversational language models and prompt engineering, Nature Communications 15 (2024) 1569. URL: https://doi.org/10.1038/s41467-024-45914-8. doi:10.1038/s41467-024-45914-8.

[120] I. Arawjo, C. Swoopes, P. Vaithilingam, M. Wattenberg, E. L. Glassman, ChainForge: A Visual Toolkit for Prompt Engineering and LLM Hypothesis Testing, in: Proceedings of the CHI Conference on Human Factors in Computing Systems, CHI '24, Association for Computing Machinery, New York, NY, USA, 2024. URL: https://doi.org/10.1145/3613904.3642016. doi:10.1145/3613904.3642016.

[121] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin, W. X. Zhao, Z. Wei, J. Wen, A survey on large language model based autonomous agents, Frontiers of Computer Science 18 (2024) 186345. URL: https://doi.org/10.1007/s11704-024-40231-1. doi:10.1007/s11704-024-40231-1.

[122] H. Zhang, W. Du, J. Shan, Q. Zhou, Y. Du, J. Tenenbaum, T. Shu, C. Gan, Building Cooperative Embodied Agents Modularly with Large Language Models, in: NeurIPS 2023 Foundation Models for Decision Making Workshop, 2023.

[123] N. Li, C. Gao, M. Li, Y. Li, Q. Liao, EconAgent: Large Language Model-Empowered Agents for Simulating Macroeconomic Activities, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 15523–15536. doi:10.18653/v1/2024.acl-long.829.

[124] Y. Qin, S. Liang, Y. Ye, K. Zhu, L. Yan, Y. Lu, Y. Lin, X. Cong, X. Tang, B. Qian, S. Zhao, R. Tian, R. Xie, J. Zhou, M. Gerstein, D. Li, Z. Liu, M. Sun, ToolLLM: Facilitating Large Language Models to Master 16000+ Real-world APIs, in: The Twelfth International Conference on Learning Representations, 2024.

[125] P. Sweetser, Large Language Models and Video Games: A Preliminary Scoping Review, in: Proceedings of the 6th ACM Conference on Conversational User Interfaces, CUI '24, Association for Computing Machinery, New York, NY, USA, 2024. URL: https://doi.org/10.1145/3640794.3665582. doi:10.1145/3640794.3665582.

[126] C. Qian, W. Liu, H. Liu, N. Chen, Y. Dang, J. Li, C. Yang, W. Chen, Y. Su, X. Cong, J. Xu, D. Li, Z. Liu, M. Sun, ChatDev: Communicative Agents for Software Development, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 15174–15186. doi:10.18653/v1/2024.acl-long.810.

[127] Y. Qin, E. Zhou, Q. Liu, Z. Yin, L. Sheng, R. Zhang, Y. Qiao, J. Shao, MP5: A Multi-modal Open-ended Embodied System in Minecraft via Active Perception, in: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024, pp. 16307–16316. doi:10.1109/CVPR52733.2024.01543.

[128] Y. Wang, Z. Jiang, Z. Chen, F. Yang, Y. Zhou, E. Cho, X. Fan, Y. Lu, X. Huang, Y. Yang, RecMind: Large Language Model Powered Agent For Recommendation, in: K. Duh, H. Gomez, S. Bethard (Eds.), Findings of the Association for Computational Linguistics: NAACL 2024, Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 4351–4364. doi:10.18653/v1/2024.findings-naacl.271.

[129] Q. Wu, G. Bansal, J. Zhang, Y. Wu, S. Zhang, E. E. Zhu, B. Li, L. Jiang, X. Zhang, C. Wang, AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation, Technical Report MSR-TR-2023-33, Microsoft, 2023. URL: https://www.microsoft.com/en-us/research/publication/autogen-enabling-next-gen-llm-applications-via-multi-agent-con

[130] A. Bajwa, S. Farooq, O. Malik, S. Khalique, H. Suguri, H. Farooq Ahmad, A. Ali, Persistent Architecture for Context Aware Lightweight Multi-agent System, in: R. H. Bordini, M. Dastani, J. Dix, A. E. F. Seghrouchni (Eds.), Programming Multi-Agent Systems, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, pp. 57–69. URL: https://link.springer.com/chapter/10.1007/978-3-540-71956-4_4.

[131] G. Li, H. A. A. K. Hammoud, H. Itani, D. Khizbullin, B. Ghanem, CAMEL: Communicative Agents for "Mind" Exploration of Large Language Model Society, in: Thirty-seventh Conference on Neural Information Processing Systems, volume 36, 2023. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/a3621ee907def47c1b952ade25c67698-Paper-Conference.pdf.

[132] R. Barbarroxa, L. Gomes, Z. Vale, Benchmarking Large Language Models for Multi-agent Systems: A Comparative Analysis of AutoGen, CrewAI, and TaskWeaver, in: P. Mathieu, F. De la Prieta (Eds.), Advances in Practical Applications of Agents, Multi-Agent Systems, and Digital Twins: The PAAMS Collection, Springer Nature Switzerland, Cham, 2025, pp. 39–48. URL: https://arxiv.org/abs/2412.21199.

[133] W. Fan, Y. Ding, L. Ning, S. Wang, H. Li, D. Yin, T.-S. Chua, Q. Li, A Survey on RAG Meeting LLMs: Towards Retrieval-Augmented Large Language Models, in: Proceedings of the 30th ACM SIGKDD Con-

ference on Knowledge Discovery and Data Mining, KDD '24, Association for Computing Machinery, New York, NY, USA, 2024, p. 6491–6501. URL: https://doi.org/10.1145/3637528.3671470. doi:10.1145/3637528.3671470.

[134] Z. Jiang, F. F. Xu, L. Gao, Z. Sun, Q. Liu, J. Dwivedi-Yu, Y. Yang, J. Callan, G. Neubig, Active Retrieval Augmented Generation, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023, Association for Computational Linguistics, 2023, pp. 7969–7992. URL: https://doi.org/10.18653/v1/2023.emnlp-main.495. doi:10.18653/V1/2023.EMNLP-MAIN.495.

[135] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, volume 33, Curran Associates, Inc., 2020, pp. 9459–9474.

[136] H. Subramonyam, R. Pea, C. Pondoc, M. Agrawala, C. Seifert, Bridging the Gulf of Envisioning: Cognitive Challenges in Prompt Based Interactions with LLMs, in: Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems, CHI '24, Association for Computing Machinery, New York, NY, USA, 2024. URL: https://doi.org/10.1145/3613904.3642754. doi:10.1145/3613904.3642754.

[137] R. Tang, Y.-N. Chuang, X. Hu, The Science of Detecting LLM-Generated Text, Commun. ACM 67 (2024) 50–59. URL: https://doi.org/10.1145/3624725. doi:10.1145/3624725.

[138] L. Fröhling, A. Zubiaga, Feature-Based Detection of Automated Language Models: Tackling GPT-2, GPT-3, and Grover, PeerJ Computer Science 7 (2021) e443.

[139] S. Thapa, U. Naseem, M. Nasim, From Humans to Machines: Can ChatGPT-like LLMs Effectively Replace Human Annotators in NLP Tasks, in: Workshop Proceedings of the 17th International AAAI Conference on Web and Social Media, 2023. URL: https://doi.org/10.36190/2023.15.

[140] S. Qi, Z. Cao, J. Rao, L. Wang, J. Xiao, X. Wang, What Is the Limitation of Multimodal LLMs? A Deeper Look into Multimodal LLMs Through Prompt Probing, Information Processing & Management 60 (2023) 103510. URL: https://www.sciencedirect.com/science/article/pii/S0306457323002479. doi:https://doi.org/10.1016/j.ipm.2023.103510.

[141] D. C. Chiang, H. Lee, Can Large Language Models Be an Alternative to Human Evaluations?, in: A. Rogers, J. L. Boyd-Graber, N. Okazaki (Eds.), Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023, Association for Computational Linguistics, 2023, pp. 15607–15631. URL: https://doi.org/10.18653/v1/2023.acl-long.870. doi:10.18653/V1/2023.ACL-LONG.870.

[142] M. Alawida, S. Mejri, A. Mehmood, B. Chikhaoui, O. Isaac Abiodun, A Comprehensive Study of ChatGPT: Advancements, Limitations, and Ethical Considerations in Natural Language Processing and Cybersecurity, Information 14 (2023). URL: https://www.mdpi.com/2078-2489/14/8/462. doi:10.3390/info14080462.

[143] W. F. Wiggins, A. S. Tejani, On the Opportunities and Risks of Foundation Models for Natural Language Processing in Radiology, Radiology: Artificial Intelligence 4 (2022) e220119. URL: https://doi.org/10.1148/ryai.220119. doi:10.1148/ryai.220119.

[144] B. D. Lund, T. Wang, Chatting About ChatGPT: How AI and GPT May Impact Academia and Libraries?, Library Hi Tech News 40 (2023) 26–29.

[145] N. Rahman, E. Santacana, Beyond Fair Use: Legal Risk Evaluation for Training LLMs on Copyrighted Text, in: ICML Workshop on Generative AI and Law, 2023.

[146] E. M. Bender, T. Gebru, A. McMillan-Major, S. Shmitchell, On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?, in: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, FAccT '21, Association for Computing Machinery, New York, NY, USA, 2021, p. 610–623. URL: https://doi.org/10.1145/3442188.3445922. doi:10.1145/3442188.3445922.

[147] M. C. Rillig, M. Ågerstrand, M. Bi, K. A. Gould, U. Sauerland, Risks and Benefits of Large Language Models for the Environment, Environmental Science & Technology 57 (2023) 3464–3466. URL: https://doi.org/10.1021/acs.est.3c01106. doi:10.1021/acs.est.3c01106, pMID: 36821477.

[148] H. Rao, Ethical and legal considerations behind the prevalence of ChatGPT: risks and regulations, Frontiers in Computing and Intelligent Systems 4 (2023) 23–29.

[149] A. Srivastava, A. Rastogi, A. Rao, A. A. M. Shoeb, A. Abid, et al., Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models, Transactions on Machine Learning Research (2023).

[150] T. Guo, X. Chen, Y. Wang, R. Chang, S. Pei, N. V. Chawla, O. Wiest, X. Zhang, Large Language Model Based Multi-agents: A Survey of Progress and Challenges, in: K. Larson (Ed.), Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24, 2024, pp. 8048–8057. URL: https://doi.org/10.24963/ijcai.2024/890. doi:10.24963/ijcai.2024/890.

[151] R. J. Neuwirth, Prohibited artificial intelligence practices in the proposed EU Artificial Intelligence Act (AIA), Computer Law & Security Review 48 (2023) 105798. URL: https://www.sciencedirect.com/science/article/pii/S0267364923000092. doi:https://doi.org/10.1016/j.clsr.2023.105798.

[152] D. Goel, F. Husain, A. Singh, S. Ghosh, A. Parayil, C. Bansal, X. Zhang, S. Rajmohan, X-Lifecycle Learning for Cloud Incident Management using LLMs, in: Companion Proceedings of the 32nd ACM International Conference on the Foundations of Software Engineering, FSE 2024, Association for Computing Machinery, New York, NY, USA, 2024, p. 417–428. URL: https://doi.org/10.1145/3663529.3663861. doi:10.1145/3663529.3663861.

[153] Y. Yang, B. Tian, F. Yu, Y. He, An Anomaly Detection Model Training Method Based on LLM Knowledge Distillation, in: 2024 International Conference on Networking and Network Applications (NaNA), 2024, pp. 472–477. doi:10.1109/NaNA63151.2024.00084.

[154] K. B. Kan, H. Mun, G. Cao, Y. Lee, Mobile-LLaMA: Instruction Fine-Tuning Open-Source LLM for Network Analysis in 5G Networks, IEEE Network 38 (2024) 76–83. doi:10.1109/MNET.2024.3421306.

[155] M. S. M. B. Shah, S. Saleem, R. Zulqarnain, Protecting digital evidence integrity and preserving chain of custody, Journal of Digital Forensics, Security and Law 12 (2017) 12.

[156] A. Wickramasekara, M. Scanlon, A Framework for Integrated Digital Forensic Investigation Employing AutoGen AI Agents, in: 2024 12th International Symposium on Digital Forensics and Security (ISDFS), 2024, pp. 01–06. doi:10.1109/ISDFS60797.2024.10527235.

[157] E. Xu, W. Zhang, W. Xu, Transforming Digital Forensics with Large Language Models: Unlocking Automation, Insights, and Justice, in: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, CIKM '24, Association for Computing Machinery, New York, NY, USA, 2024, p. 5543–5546. URL: https://doi.org/10.1145/3627673.3679091. doi:10.1145/3627673.3679091.

[158] M. M. Al Mahdi, S. Baror, Proof of Concept of a Digital Forensic Readiness Cybercrime Language as a Service, in: International Conference on Cyber Warfare and Security, volume 19, 2024, pp. 191–199. doi:https://doi.org/10.34190/iccws.19.1.2059.

[159] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, et al., Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, in: Advances in Neural Information Processing Systems, volume 33, Curran Associates, Inc., 2020, pp. 9459–9474. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf.

[160] S. Shafee, A. Bessani, P. M. Ferreira, Evaluation of LLM-based chatbots for OSINT-based Cyber Threat Awareness, Expert Systems with Applications 261 (2025) 125509. URL: https://www.sciencedirect.com/science/article/pii/S0957417424023765. doi:https://doi.org/10.1016/j.eswa.2024.125509.

[161] N. M. Karie, V. R. Kebande, H. Venter, K.-K. R. Choo, On the Importance of Standardizing the Process of Generating Digital Forensic Reports, Forensic Science International: Reports 1 (2019) 100008. doi:https://doi.org/10.1016/j.fsir.2019.100008.

[162] C. Champod, A. Biedermann, J. Vuille, S. Willis, J. De Kinder, ENFSI guideline for evaluative reporting in forensic science: A primer for legal

practitioners, Criminal Law and Justice Weekly 180 (2016) 189–193.
[163] A. Valjarević, H. Venter, R. Petrović, Iso/iec 27043: 2015—role and application, in: 2016 24th Telecommunications Forum (TELFOR), IEEE, 2016, pp. 1–4.
[164] G. Michelet, F. Breitinger, ChatGPT, Llama, can you write my report? An experiment on assisted digital forensics reports written using (local) large language models, Forensic Science International: Digital Investigation 48 (2024) 301683. URL: https://www.sciencedirect.com/science/article/pii/S2666281723002020. doi:https://doi.org/10.1016/j.fsidi.2023.301683, DFRWS EU 2024 - Selected Papers from the 11th Annual Digital Forensics Research Conference Europe.
[165] F. Breitinger, J.-N. Hilgert, C. Hargreaves, J. Sheppard, R. Overdorf, M. Scanlon, DFRWS EU 10-Year Review and Future Directions in Digital Forensic Research, Forensic Science International: Digital Investigation (2024).
[166] S. Dai, C. Xu, S. Xu, L. Pang, Z. Dong, J. Xu, 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, KDD '24, Association for Computing Machinery, New York, NY, USA, 2024, p. 6437–6447. URL: https://doi.org/10.1145/3637528.3671458. doi:10.1145/3637528.3671458.
[167] R. Zhou, Empirical Study and Mitigation Methods of Bias in LLM-Based Robots, Academic Journal of Science and Technology 12 (2024) 86–93. URL: https://doi.org/10.54097/re9qp070. doi:10.54097/re9qp070.
[168] J. Mökander, J. Schuett, H. R. Kirk, L. Floridi, Auditing large language models: a three-layered approach, AI and Ethics (2023). URL: https://doi.org/10.1007/s43681-023-00289-2. doi:10.1007/s43681-023-00289-2, online first.
[169] M. Scanlon, B. Nikkel, Z. Geradts, Digital forensic investigation in the age of ChatGPT, Forensic Science International: Digital Investigation 44 (2023) 301543. doi:https://doi.org/10.1016/j.fsidi.2023.301543.
[170] Y. Yao, J. Duan, K. Xu, Y. Cai, Z. Sun, Y. Zhang, A Survey on Large Language Model (LLM) Security and Privacy: The Good, The Bad, and The Ugly, High-Confidence Computing 4 (2024) 100211. URL: https://www.sciencedirect.com/science/article/pii/S266729522400014X. doi:https://doi.org/10.1016/j.hcc.2024.100211.
[171] H. Brown, K. Lee, F. Mireshghallah, R. Shokri, F. Tramèr, What Does it Mean for a Language Model to Preserve Privacy?, in: Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency, FAccT '22, Association for Computing Machinery, New York, NY, USA, 2022, p. 2280–2292. URL: https://doi.org/10.1145/3531146.3534642. doi:10.1145/3531146.3534642.
[172] J. Zou, S. Zhang, M. Qiu, Adversarial Attacks on Large Language Models, in: C. Cao, H. Chen, L. Zhao, J. Arshad, T. Asyhari, Y. Wang (Eds.), Knowledge Science, Engineering and Management, Springer Nature Singapore, Singapore, 2024, pp. 85–96. URL: https://link.springer.com/chapter/10.1007/978-981-97-5501-1_7.
[173] Z. Ji, T. Yu, Y. Xu, N. Lee, E. Ishii, P. Fung, Towards Mitigating LLM Hallucination via Self Reflection, in: H. Bouamor, J. Pino, K. Bali (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2023, Association for Computational Linguistics, Singapore, 2023, pp. 1827–1843. doi:10.18653/v1/2023.findings-emnlp.123.
[174] G. Perković, A. Drobnjak, I. Botički, Hallucinations in LLMs: Understanding and Addressing Challenges, in: 2024 47th MIPRO ICT and Electronics Convention (MIPRO), 2024, pp. 2084–2088. doi:10.1109/MIPRO60963.2024.10569238.
[175] Y. Yehuda, I. Malkiel, O. Barkan, J. Weill, R. Ronen, N. Koenigstein, InterrogateLLM: Zero-Resource Hallucination Detection in LLM-Generated Answers, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 9333–9347. doi:10.18653/v1/2024.acl-long.506.
[176] Z. Xu, F. Jiang, L. Niu, J. Jia, B. Y. Lin, R. Poovendran, SafeDecoding: Defending against Jailbreak Attacks via Safety-Aware Decoding, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 5587–5605. doi:10.18653/v1/2024.acl-long.303.