Forensic Science International: Digital Investigation 56 (2026) 302052

Contents lists available at ScienceDirect

Forensic Science International: Digital Investigation

journal homepage: www.elsevier.com/locate/fsidi

DFRWS EU 2026 - Selected Papers from the 13th Annual Digital Forensics Research Conference Europe

Hey GPT-OSS, looks like you got it – Now walk me through it! An assessment of the reasoning language models chain of thought process for digital forensics

Gaëtan Michelet [*], Janine Schneider, Aruna Withanage, Frank Breitinger [**]

Cybersecurity, University of Augsburg, 86159, Augsburg, Germany

ARTICLE INFO

Keywords:
Digital forensics investigation
Large language models
Local language models
Gpt-oss
Reasoning language models
Chain of thought mechanism

ABSTRACT

Large language models (LLMs), including systems such as ChatGPT, are increasingly examined for their role in digital forensics. Current research not only surveys their potential applications but also investigates how fine-tuning and model adaptation can enhance performance on specialized forensic tasks. However, the understandability and interpretability of the results (outputs) reduce their operational and legal usability. Recently, a new class of reasoning language models has emerged, designed to handle logic-based tasks through an 'internal reasoning' mechanism. Yet, users typically only see the final answer, not the underlying reasoning. One of these reasoning models is gpt-oss, which can be deployed locally, providing full access to its underlying reasoning process. This article presents the first investigation into the potential of reasoning language models for digital forensics. Four test use cases are examined to assess the usability of the reasoning component in supporting results understandability. The evaluation combines a new quantitative metric with qualitative analysis. Findings show that the reasoning component aids in understanding, interpreting, and validating LLM outputs in digital forensics at medium reasoning levels, but the support is often limited, and higher reasoning levels do not enhance response quality.

1. Introduction

Researching Artificial Intelligence is important for digital forensics practitioners (Hargreaves et al., 2024). Consequently, large language model (LLM) capabilities have been explored for various tasks, e.g., their ability to write scripts related to digital forensics (Wickramasekara et al., 2025b), to help with volatility-based memory forensics (Oh et al., 2024; Lang and Schreck, 2025), and to support forensic report writing (Michelet and Breitinger, 2024).

A challenge with LLMs is understanding (and thus explaining) how these models derive the results they return to the user. Consequently, there is a need to clarify and assess LLM-generated responses, especially when they are used during an investigation, as their outputs may need to be explained and justified in court. Improving LLM interpretability requires insight into the steps a model takes to generate its answers, thereby supporting the verification of its outputs. Recent Reasoning Language Models (RLMs), including those from OpenAI, implement a Chain-of-Thought mechanism designed to reveal such internal reasoning. This new class of LLMs incorporates a step-by-step reasoning layer that enhances their ability to solve complex tasks such as arithmetic or coding problems. In essence, the model first produces an internal sequence of reasoning steps before generating the final answer presented to the user. While the term Chain of Thought (CoT) is often used to describe a prompting technique (see Sec. 2), OpenAI also uses it to refer to the internal reasoning process of gpt-oss. Throughout this paper, CoT refers to this internal reasoning process unless explicitly linked to prompting. The terms reasoning process, CoT, and reasoning component are used interchangeably.

While the term reasoning is used, RLMs do not actually think or reason. Similarly to regular language models, they predict one of the most probable next tokens given the current context, and thus in an iterative way, i.e., they generate one token after another, appending the last generated token to the context. This ability to use *reasoning* is due to the training process, which tries to mimic human reasoning. As the CoT

* Corresponding author.
** Corresponding author.https://www.FBreitinger.de
E-mail addresses: gaetan.michelet@uni-a.de (G. Michelet), janine.schneider@uni-a.de (J. Schneider), aruna.withanage@uni-a.de (A. Withanage), frank.breitinger@uni-a.de (F. Breitinger).

https://doi.org/10.1016/j.fsidi.2026.302052

is also 'guessed' by the model, we decided to move away from the terms *explainability* and *justifiability* (often associated with explainable AI, which is out of scope for this paper). Consequently, this article uses the terms: understandability, interpretability, and verifiability.

Current common reasoning models are Deepseek's Deepseek-R1 and Qwen's QwQ, which are freely accessible and locally deployable RLMs.[1][2] They were released in January 2025 and March 2025, respectively. OpenAI released gpt-oss,[3] an RLM offered with 20 or 120 billion parameters, in August 2025. The smaller version is supposed to be useable on a 16 GB GPU.

All three models can be deployed and operated locally, making them well-suited for investigations by eliminating data sharing with external AI providers and enabling direct access to the model's reasoning output. Once this additional part of the generated text is extracted (the reasoning process is usually hidden from the user), the elements it contains may serve as supporting details, i.e., enhancing the comprehensibility, interpretability, and verifiability of LLM-generated results and thereby increasing the trustworthiness and reliability of the forensic process. This paper investigates the potential and impact of RLMs for digital forensics and focuses on the following research question:

To what extent can the result from the chain of thought mechanism of reasoning language models help to understand and interpret the obtained results when undertaking digital forensics tasks?

To answer this question, we conduct experiments using OpenAI's gpt-oss-20b model, which can run locally, and provide access to the reasoning section. In addition, it is one of the newest publicly released reasoning models, and its balanced size makes it straightforward to deploy.

We created four test scenarios and evaluated both the reasoning process as well as the final answer for each of the scenarios. To evaluate the results, we decided to develop the following secondary research questions:

RQ1 What is the quality of the CoT generated by gpt-oss for a given digital forensics task?
RQ2 How does the adjustable level of CoT impacts the quality of the reasoning section?
RQ3 Does the quality of the CoT impact the quality of the final answer?
RQ4 Can the CoT effectively assist in understanding why the model reached a certain conclusion?

1.1. Contribution

To the best of our knowledge, we are the first to investigate the potential of RLMs as 'supporting details' to improve LLM-understandability for forensic tasks. By doing so, we developed four unique testing scenarios, adapted standard RLM evaluation metrics to evaluate the model's CoT quality, introduced additional metrics to assess the final responses' quality, evaluated the usability of OpenAI's gpt-oss model's CoT through quantitative metrics and qualitative evaluation techniques, and examined the impact of different factors on the quality of the CoT and final response.

1.2. Outline

The remainder of this paper is structured as follows: We present the background on RLMs and important related work in Sec. 2. This is followed by the description of the methodology in Sec. 3. The methodology section includes the description of the four testing scenarios and the testing environment. It also contains details on the setup of the experiments, important parameters, and the description of the evaluation process. Next, we present the results of our study in Sec. 4, the discussion of the results in Sec. 5, and limitations and future work in Sec. 6. We conclude the paper in Sec. 7.

2. Background and related work

2.1. Large language models in digital forensics

There is a fair amount of research on using LLMs in digital forensics, although the field is still maturing. Note that the investigation of artifacts left behind by the use of LLMs on a system is out of scope for our study. This section therefore reviews the existing research on applying LLMs within the digital forensics process.

Several articles focused on the application of LLMs to a specific forensic task or a set of tasks. For example, Henseler and van Beek (2023) investigated the capabilities of LLMs for three use cases: generating queries for Hansken, summarizing, evaluating, and visualizing electronic communications, and analyzing search outputs. Michelet and Breitinger (2024) examined the potential of LLMs, such as ChatGPT and Llama-2, to support the writing of forensic reports. Wickramasekara et al. (2025b) introduced AutoDFBench, an automated framework for testing and benchmarking AI-generated digital forensic code, including outputs from LLMs.

LLM applicability has also been explored in memory forensics, where Oh et al. (2024) introduced volGPT, an approach for ransomware triage that integrates an LLM with the Volatility framework. Similarly, Lang & Schreck (2025) examined the incorporation of LLMs into memory forensics workflows to improve the detection of stealthy malware and advanced persistent threats.

The capabilities, limits, and risks associated with the use of LLMs in digital forensics have also been discussed: Scanlon et al. (2023a) evaluated the capabilities and limitations of ChatGPT-4 in a range of forensic contexts. Scanlon et al. (2023b) discussed the transformative impact of LLMs on digital forensics, reflecting on how the rapid adoption of such models has sparked promising applications, but also general debates about trust, authenticity, and reliability when using LLMs in various domains. Dinis-Oliveira and Azevedo (2023) explored how ChatGPT could serve as a virtual assistant for lawyers, judges, and victims to interpret and manage digital forensics expert evidence.

The broader research landscape additionally encompasses studies on LLM optimization, including fine-tuning approaches. For instance, Sharma et al. (2025) presented ForensicLLM, a fine-tuned local LLaMA-3.1–8B model specifically adapted for forensics tasks and trained with question and answer pairs extracted from research articles and curated digital artifacts.

Practical recommendations for fine-tuning LLMs for digital forensics tasks were developed by Michelet et al. (2025). To demonstrate the applicability of their approach, the authors presented a case study on chat summarization, evaluating the performance of several fine-tuned models. Finally, Cho et al. (2024) explored the use of LLMs and fine-tuning for author profiling in digital text forensics, e.g., trying to detect the gender or age of an author based on the writing style.

Further discussion on how LLMs can be applied is provided by Wickramasekara et al. (2025a), who explored the use of LLMs for different aspects of digital forensics, and Xu et al. (2025), who reviewed different integrations of LLMs to the digital forensics process. Despite extensive research on LLMs in digital forensics, prior work does not investigate their reasoning capabilities.

2.2. Reasoning and large language models

Before the creation of specialized reasoning models, researchers used few-shot prompting and prompt engineering to simulate and obtain the

[1] https://www.huggingface.co/deepseek-ai/DeepSeek-R1.
[2] https://www.huggingface.co/Qwen/QwQ-32B.
[3] https://www.huggingface.co/collections/openai/gpt-oss-68911959590a1634ba11c7a4.

'thought process' of LLMs. Few-shot prompting consists of guiding the model by providing examples showing how the model is expected to achieve the task: it often includes a few input–output examples embedded in the prompt. For example, Wei et al. (2022) investigated how a few-shot prompting combined with a chain of thought can enhance the reasoning capabilities of LLMs. Their experiments demonstrated that this approach enables LLMs to solve complex tasks in arithmetic, commonsense, and symbolic reasoning more effectively than standard prompts. Other methods are presented by Plaat et al. (2025), who reviewed different approaches that leverage LLMs' reasoning capabilities through prompting.

Creswell and Shanahan (2022) designed a workflow producing faithful multi-step reasoning using LLMs. Their method integrates reasoning steps generated by two fine-tuned LLMs, independently handling the selection and inference processes. The selection model identifies potentially relevant facts from the context, i.e., the base text provided to the model and extended during inference, and supplies them to the inference model. The inference model then reasons over this selected content. Its generated output is appended to the context, forming one step in the reasoning process. This procedure iterates until the question is fully answered.

Since the rise of these reasoning process ideas, different surveys have been published, summarizing the state of the art of reasoning and LLMs. For example, Patil and Jadon (2025) provided a comprehensive survey focusing on enhancing the reasoning capabilities in LLMs, and Huang & Chen-Chuan Chang (2023) provided a comprehensive review targeting the LLMs' reasoning capabilities.

2.3. Reasoning language models

A regular LLM is trained to predict the next token based on the current context. Its reasoning abilities emerge implicitly, meaning the model can solve problems that require logical steps, but it is not explicitly trained to do so or to display the reasoning process. RLMs, on the other hand, are specifically trained to produce explicit reasoning steps before providing a final answer. These intermediate steps are sometimes called a chain of thought (CoT). When an RLM receives a command prompt, it breaks the task into smaller steps and tries to solve them sequentially. After generating these reasoning steps, the model produces the final answer based on its intermediate reasoning. Consequently, the CoT mechanism helps models perform more accurately on tasks that require reasoning, while also making their internal thought processes more transparent and interpretable. More details on the RLM components are presented by Besta et al. (2025), who surveyed RLMs' related work.

While studies often focus on evaluating the final answer by using benchmark datasets, some work mentions ways to evaluate the reasoning process itself. It is sometimes an element of the survey (Patil and Jadon, 2025; Plaat et al., 2025; Huang & Chen-Chuan Chang, 2023), and sometimes the target of the survey (Mondorf and Plank, 2024).

More recently, Lee and Hockenmaier (2025) highlighted inconsistencies in the reasoning evaluation process and pointed out the variability of available metrics based on the reasoning task at hand. They proposed a structured taxonomy with four criteria that can be applied to any reasoning task: factuality, validity, coherence, and utility. These four criteria will be utilized within this study.

2.4. LLM/RLM specifics

This section provides background knowledge on LLMs and RLMs, which is essential to understanding our experiment.

2.4.1. Chat templates and prompts

Chat templates are pre-defined structures for interacting with LLMs. They typically specify roles (like 'assistant' or 'user') and provide contextual instructions. The main purpose of templates is to guide the model's behavior, ensuring consistent, coherent, and task-appropriate responses. Using templates allows users to control tone, style, and output type without modifying the model itself, making interactions more predictable and efficient. To utilize a chat template to generate effective prompts, the template's pre-defined fields must be populated with task-specific instructions and contextual information. Fig. 1 shows an example of the gpt-oss chat template used in our experiments.

2.4.2. RLM parameters

RLMs rely on several parameters that influence the quality and reliability of the reasoning process. For our study, the following parameters are relevant:

- The *maximum number of tokens* generated sets an upper bound on the model's output length; if it is too low, the inference may be truncated, whereas overly large values can lead to redundant or unfocused outputs.
- The *sampling strategy* determines how tokens are selected during text generation. Deterministic methods produce consistent but rigid results, while probabilistic methods introduce controlled randomness that can enhance reasoning diversity and creativity.
- The *temperature parameter* further adjusts this randomness by shaping the probability distribution. Low values lead to deterministic, precise reasoning, while higher values yield more exploratory but potentially inconsistent results.

In our experiments, we determined the parameters in advance through pre-testing. Furthermore, the sampling of the generated tokens and the temperature vary between the different experimental setups as described in Sec. 3.1.

3. Methodological framework and experimental design

We conducted four experiments comprising multiple runs with varying parameter configurations to systematically analyze gpt-oss' behavior under different conditions. In total, we conducted and evaluated 60 experiments. The following subsections describe the setup, parameter choices, and evaluation procedures in detail, ensuring

```
<|start|>system<|message|>ROLE
Knowledge cutoff: 2024-06
Current date: DATE

Reasoning: REASONING LEVEL

# Valid channels: analysis, commentary, final.
Channel must be included for every message.
<|end|>

<|start|>user<|message|>PROMPT
<|end|>

<|start|>assistant<|channel|>analysis<|message|>CoT
<|end|>

<|start|>assistant<|channel|>final<|message|>ANSWER
<|return|>
```

Fig. 1. Chat template for the gpt-oss reasoning language model (simplified and arranged for readability). Green text represents contextual information submitted to the model, i.e., the system information. ROLE (sometimes referred to as the model identity) and REASONING LEVEL (low, medium, or high) are manually set, while DATE is automatically computed when the template is applied. Text depicted in blue shows the manually created user prompt. The orange text represents the model's inference, i.e., the text that the model will generate. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

transparency and reproducibility of the results.

3.1. Experimental setup

For our study, we simulated four typical tasks:

- Suspicious Message Detection (SMD): The model is provided with information about a case investigated and messages extracted from a chat conversation. It must then determine whether the chat is relevant to the investigation or not.
- Bash History Analysis (BHA): The model is supplied with the Bash history extracted from a laptop belonging to an individual suspected of attempting to attack a server. It must determine if an attack was undertaken from that laptop and, if so, who was the target.
- Methodology Generation (MG): The model receives information about a case under investigation and the associated investigative question. It must then devise a methodology capable of addressing that question.
- Timeline Analysis (TA): The model is provided with information about a suspected data theft and a timeline obtained from a forensic image. It must then determine whether the timeline data can confirm the theft and determine if the internet was active during that time. Note, the timeline contains a USB drive connection and artifacts that are potentially related to an active internet connection.

Further details about the tasks and the full prompts can be found on GitHub.[4]

The tasks were chosen based on three criteria: in practice, they must be burdensome (i.e., occur frequently or be time-consuming) to perform manually, require a certain form of reasoning, and the input data for the task must be in a format supported by gpt-oss. For each task, a prompt is generated using the gpt-oss chat template with the following values:

- ROLE: "You are a large language model specialized in digital forensics investigation." was used as the model role for each run.
- REASONING LEVEL: Low, Medium, and High, depending on the current run.
- PROMPT: Designed to focus on the context, the input data, the instructions, and the output format.

For each task, one prompt is generated that is used under different conditions, i.e., with various parameters, as depicted in Table 1. We have a total of 60 tests.

Note that while using probabilistic sampling adds randomness to the generation, not using sampling produces consistent results. Therefore, generating several texts without sampling is not required, as they will always be identical. However, generating several texts with sampling enabled allows us to evaluate how the model performs when randomness is included in the generation process. When using sampling during the generation, we used a temperature of 0.7. This value was included in a code snippet provided by OpenAI showcasing how to generate text using gpt-oss and the generate()-function of the transformers library.[5]

Inferences were run on the cluster provided by the authors' institution. The cluster's resources are: 1 AMD EPYC-7713 CPU with 4 GB of RAM and 1 Nvidia A100 with 80 GB of VRAM.

3.2. Evaluation

For each experiment, both the reasoning process and the final answer are examined to assess the model's performance and its potential to improve the understandability of LLM-generated results in the context of digital forensics. The analysis focuses on how different parameter settings influence the quality, coherence, and reliability of the generated reasoning as well as the correctness of the resulting outputs. We employ quantitative metrics derived from established measures in the literature, alongside qualitative evaluation techniques. The evaluation of the results was split among three evaluators in a 2/1/1 ratio (i.e., 30, 15, and 15 samples). A meeting was organized between the evaluators to discuss the metrics and how the CoT should be evaluated. Given the exploratory nature of the study, we chose to work with a smaller set of samples to maximize control over the evaluation. For this reason, we opted for manual assessment.

Since we are interested in the reasoning process's potential to make the final answer comprehensible, we first evaluate the CoT. For that, we decided to follow the four generic metrics presented by Lee and Hockenmaier (2025):

Factuality is evaluated using the number of factual errors in the reasoning process. A factual error is a fact incorrectly taken from the prompt or derived from incorrect common knowledge. An example of a factual error would be if the model refers to a specific date in the given timeline that is actually not included in the data.

Coherence is evaluated similarly, but instead of the number of factual errors, the number of incoherences is computed. An incoherence is present when the model makes an inference while the preconditions for that inference have not yet been presented in the CoT. An example of an incoherence would be: "The chat is about Breaking Bad" without previously mentioning elements from the chat that relate to the TV show.

Validity is evaluated using the number of logical/reasoning errors present in the reasoning process. A logical/reasoning error is a mistake in an inference that is due to a logical problem. An example of a logical error would be: "We don't see a file copy event. The file was probably copied."

Utility is evaluated using the number of unnecessary steps, i.e., if it does not help the model to answer the question. An example of an unnecessary step is repeating the task without adding any new aspects to the following reasoning.

Table 1
Details on the different experiments conducted in this study. SMD refers to the Suspicious Message Detection experiments, BHA refers to the Bash History Analysis experiments, MG refers to the Methodology Generation experiments, and TA refers to the Timeline Analysis experiments.

ID	Task	Reasoning	Temperature	Sampling	Repetitions
1–4	SMD	Low	0.7	Yes	4
5	SMD	Low	0	No	1
6–9	SMD	Medium	0.7	Yes	4
10	SMD	Medium	0	No	1
11–14	SMD	High	0.7	Yes	4
15	SMD	High	0	No	1
16–19	BHA	Low	0.7	Yes	4
20	BHA	Low	0	No	1
21–24	BHA	Medium	0.7	Yes	4
25	BHA	Medium	0	No	1
26–29	BHA	High	0.7	Yes	4
30	BHA	High	0	No	1
31–34	MG	Low	0.7	Yes	4
35	MG	Low	0	No	1
36–39	MG	Medium	0.7	Yes	4
40	MG	Medium	0	No	1
41–44	MG	High	0.7	Yes	4
45	MG	High	0	No	1
46–49	TA	Low	0.7	Yes	4
50	TA	Low	0	No	1
51–54	TA	Medium	0.7	Yes	4
55	TA	Medium	0	No	1
56–59	TA	High	0.7	Yes	4
60	TA	High	0	No	1

[4] https://github.com/Michelet-Gaetan/Hey-GPT-OSS-Looks-Like-You-Got-It-Now-Walk-Me-Through-It.

[5] https://cookbook.openai.com/articles/gpt-oss/run-transformers#advanced-inference-with-generate.

Besides that, we count the number of repetition loops in which the model gets stuck occasionally. A repetition loop is an ongoing repetition of a word, a sequence of words, a sentence, or a sequence of sentences repeated twice or more. In that case, the whole loop is considered a single step and is classified as one repetition.

We use these metrics to answer Research Question 1 and Research Question 2. To answer Research Question 3, we evaluate the correctness of the final answer and the quality of the justification given for each answer.

Correctness is evaluated by comparing the expected answer components for each experiment with the provided answer. Each expected component is explicitly defined and must be correctly reflected in the model response. For tasks such as SMD, only a single component is expected. For more complex tasks, the expected answers consist of two to eight components.[6] Note that for the methodology generation, unrequested components present in the answer were penalized if incorrect.
The quality of the **Justification** is evaluated qualitatively by the evaluator and converted to 1 or 0 (satisfying or not) to calculate a numerical value.

We compute separate scores for the CoT and the final answer, which we then use to compare the different experiments and parameters and to answer our research questions.

The **CoT Score** is the average of the normalized metric values introduced earlier. We normalized each CoT metric value by taking into account the number of steps in the reasoning process. These normalized values represent inverted metrics, as they indicate the relative proportion of steps that did not satisfy the metric, e.g., the relative number of factual errors is used to assess factuality. To obtain the expected metric, the normalized values are subtracted from 1: $1 - (\text{NBelements}/\text{NBsteps})$.
The **Final Answer Score** is a weighted combination of the correctness and justification values. Correctness, as described earlier, is normalized by the number of expected correct predefined answer components. The final answer score is a weighted mean, where correctness is counted three times and justification once.

Note, the number of steps was determined manually by the evaluators, as one reasoning step can potentially comprise more than one sentence. The number of steps should not be confused with the number of tokens, which can be determined automatically.

4. Results

The findings highlight how variations in parameter settings affect both the model's reasoning behavior and the accuracy of its final outputs. Key trends and notable differences across experiments are summarized to provide a basis for the subsequent discussion.

Unfortunately, two (out of 60) experiments could not be analyzed as they reached the limit of 12'500 new tokens that we decided to set for the text generation[7]. We selected this limit based on an estimate of the text required to complete the chosen tasks and on several preliminary experiments. These two samples are excluded from the results analysis, except when stated otherwise.

4.1. Quality of the reasoning process and the final answer

This section discusses the quality of the reasoning process, the resulting CoT, and the quality of the final answer.

Fig. 2 presents the average metric values across all experiments, as well as the average values for each individual task. Among the four tasks, methodology generation and timeline analysis showed lower correctness, which can be attributed to the higher complexity of these tasks. In the case of MG, the model often produced useless additional responses that were not asked for, e.g., the model frequently included legal aspects in the response. In the case of TA, the model had to choose between a yes or no answer despite the sparse data available. In all cases, the model decided to go with the suspicion mentioned in the context, even though there was no clear evidence for it. These two tasks also have the highest number of steps and tokens generated for the CoT, indicating a higher level of complexity.

The factuality is overall good, but lower for MG. This can be explained by the fact that the MG had to use a lot of common digital forensics knowledge in the reasoning process, which was often inaccurate. On the other hand, the three other tasks relied on the data provided in the prompt and therefore made fewer factual errors.

Regarding the coherence, two outliers can be detected: the MG and the BHA. For the former, incoherences were present when the model drafted the methodology structure by adding new steps or components that were not previously mentioned. The latter often provided answers first and then explained them, which is not the correct reasoning order.

Exact repetitions were detected in texts generated for the MG and SMD. For the MG, the final answer score was also generally better than the pure correctness value, as the final answer was often well justified, even when not fully accurate. On the other hand, answers were often not correctly justified for the timeline analysis, explaining why the final score is worse than the pure correctness value.

Therefore, Research Question 1 can be answered as follows: The quality of the generated CoTs differs among the different forensic tasks. However, the CoT generally appears to be of good quality and useful.

4.2. Impact of the reasoning level on the CoT quality

We now discuss the impact of the reasoning level on the quality of the generated CoT.

Fig. 3 presents the average value of each metric across the different reasoning levels. Each row represents the mean metric values for the samples generated at a given reasoning level.

Overall, the CoT score is highest at the medium reasoning level. This can be explained by an increase in coherence and a decrease in utility as the reasoning level rises. Higher reasoning levels produce longer CoTs

Fig. 2. Averaged metric values for all experiments combined and for all experiments of each separate task. The 'general' row represents the average value of each metric across all samples. In contrast, each task represents the average metric values calculated over the samples specific to that task. The closer a metric value is to one, the better it is.

[6] These expected answer components are detailed in the experiment GitHub: https://www.github.com/Michelet-Gaetan/Hey-GPT-OSS-Looks-Like-You-Got-It-Now-Walk-Me-Through-It.

[7] Both of them were generated using the deterministic method and had a high reasoning level setting

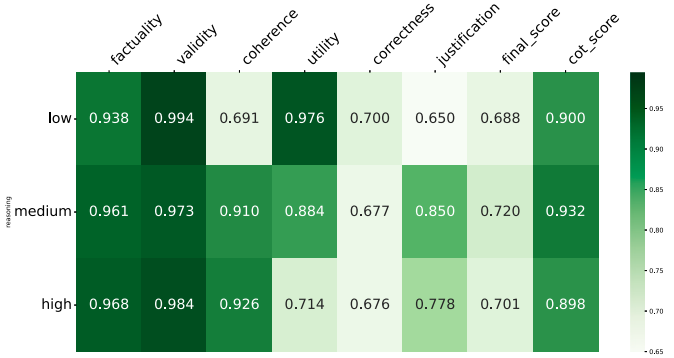

Fig. 3. Averaged metric values for each reasoning level.

with more steps and tokens. When the number of steps is too small, the model fails to establish the necessary preconditions, which reduces coherence. Conversely, when the number of steps becomes too large, the model tends to repeat content or explore irrelevant possibilities, thereby lowering utility. The medium reasoning level appears to balance coherence and utility well.

Additional metric values were computed for each task, revealing that the coherence trend differs for MG and TA. Both tasks achieve the highest coherence at the medium reasoning level. This can be explained by the TA task's tendency to produce overly sophisticated hypotheses not grounded in the input data. On the other hand, MG generated a larger number of methodology drafts at the medium and high reasoning levels, elements of the reasoning process that exhibited the greatest incoherence. Finally, the BHA also exhibited a difference, showing the best CoT score with the high reasoning level. This might be explained by a significant improvement in validity from the medium to the high reasoning level for this task.

Therefore, Research Question 2 can be answered as follows: The reasoning level directly impacts the quality of the CoT, with the medium level of reasoning resulting in the best quality for most tasks.

4.3. Impact of the reasoning process on the final answer

We also evaluated the impact of the reasoning process (the model's ability to 'think') on the quality of the final answer.

When plotting the CoT score against the final answer scores, no clear trend or observable effect can be identified, as shown in Fig. 4. The plot shows the score obtained for the reasoning process on the x-axis and the score obtained for the final answer on the y-axis for all experiments. The higher the score, the better the evaluated element. This absence of a

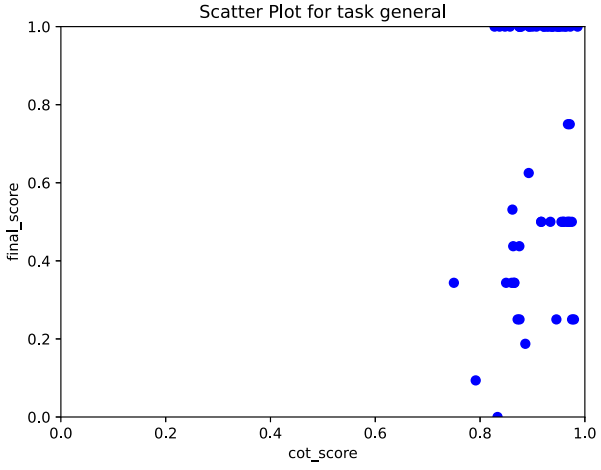

Fig. 4. Scatter plot showing the score of the reasoning process (CoT) on the x-axis and the score of the final answer (final) on the y-axis for all experiments.

trend or relation could be interpreted as the absence of an impact of the reasoning process on the quality of the final answer. The quality of the final answer seems to be influenced by the complexity of the task.

For Research Question 3, we can conclude: No, the quality of the CoT does not directly impact the quality of the final answer, given the scope of our experiment.

4.4. Impact of the generation strategy

During the inferences, two types of generation were used: sampling (with a temperature of 0.7) and a deterministic approach (similar to a temperature of 0). Most of the exact repetitions were observed in the texts generated using the deterministic approach.

These deterministically generated texts also had a lower final score, a higher CoT score, a smaller number of reasoning steps, and a higher number of tokens generated for the reasoning process than texts generated with sampling. This could be explained by the higher number of repetitions, which increases the number of tokens in the CoT. The behavior was different for TA and BHA, two tasks without repetition.

To provide an overview of the results, we listed them in detail in Table 2.

4.5. Qualitative results

Finally, we discuss the qualitative evaluation results for each of the four digital forensics test scenarios.

4.5.1. Suspicious message detection

In the context of the SMD task, the reasoning patterns across samples show notable consistency. Samples usually start in a similar way: by assuming that the chat refers to the Breaking Bad TV show and then evaluating its relevance. The final answers are consistently correct and well justified.

CoTs produced with low reasoning levels, while easier to read and follow, do not provide sufficient content to better understand the final answer. In contrast, the medium and high reasoning levels allow for more extensive reasoning and offer a better support to comprehend the final answer, though both also contain more repetitive content.

4.5.2. Bash history analysis

In the server attack investigation task context, the model's reasoning process follows a recurring pattern. The model often provides an initial answer to the question at the very beginning of the reasoning process, relying on only a few elements. It then proceeds to expand the evidence list or explore alternative hypotheses. The reasoning sequence is often repetitive: the model returns to the question, answers it, introduces additional evidence or tests new hypotheses, and then revisits the question.

References to the commands included in the input data are generally accurate. However, the model occasionally has difficulties with the attack flow; for instance, it confuses a reverse shell with a shell.php file downloaded from the targeted server. Despite these issues, the final answers are consistently correct and well justified.

CoTs produced at the low reasoning level are insufficient to better understand the final answer, as the response is given directly without any reasoning or supporting evidence. The medium reasoning level introduces more elements, but may still not be sufficient. In contrast, the high reasoning level is likely sufficient to understand all aspects, although it contains considerably more repetitive content.

4.5.3. Methodology generation

The reasoning process generated during the MG task differs slightly from the others. At the beginning, the model reiterates and elaborates on what needs to be done, but soon shifts into producing a list of elements to be included in the final answer.

Samples generated with the high reasoning level, and one of the

Table 2
Overall metric values for all experiments, including the average values for all experiments and per task.

ID	Task	Reasoning	Sampling	Factuality	Validity	Coherence	Utility	CoT Score	Correctness	Justification	Answer Score
Mean	General	–	–	0.955	0.984	0.839	0.863	0.910	0.685	0.759	0.703
1–4	SMD	Low	Yes	1.000	1.000	0.775	1.000	0.944	1.000	1.000	1.000
5	SMD	Low	No	1.000	1.000	0.800	1.000	0.950	1.000	1.000	1.000
6–9	SMD	Medium	Yes	1.000	1.000	0.934	0.920	0.963	1.000	1.000	1.000
10	SMD	Medium	No	1.000	1.000	0.944	1.000	0.986	1.000	1.000	1.000
11–14	SMD	High	Yes	1.000	1.000	0.936	0.454	0.847	1.000	1.000	1.000
15	SMD	High	No	1.000	1.000	0.957	0.617	0.894	1.000	1.000	1.000
Mean	SMD	–	–	1.000	1.000	0.886	0.807	0.923	1.000	1.000	1.000
16–19	BHA	Low	Yes	1.000	1.000	0.482	1.000	0.871	1.000	1.000	1.000
20	BHA	Low	No	1.000	1.000	0.600	1.000	0.900	1.000	1.000	1.000
21–24	BHA	Medium	Yes	0.991	0.929	0.773	0.929	0.906	1.000	1.000	1.000
25	BHA	Medium	No	1.000	0.931	0.793	1.000	0.931	1.000	1.000	1.000
26–29	BHA	High	Yes	0.985	0.967	0.908	0.853	0.928	1.000	1.000	1.000
30	BHA	High	No	–	–	–	–	–	–	–	–
Mean	BHA	–	–	0.993	0.965	0.718	0.938	0.903	1.000	1.000	1.000
31–34	MG	Low	Yes	0.789	1.000	0.520	1.000	0.827	0.250	0.750	0.375
35	MG	Low	No	0.667	1.000	0.667	1.000	0.833	0.000	0.000	0.000
36–39	MG	Medium	Yes	0.837	0.982	0.934	0.704	0.865	0.219	0.750	0.352
40	MG	Medium	No	0.913	1.000	0.913	0.957	0.946	0.000	1.000	0.250
41–44	MG	High	Yes	0.899	0.976	0.867	0.736	0.869	0.063	1.000	0.297
45	MG	High	No	0.893	0.987	0.927	0.645	0.863	0.250	1.000	0.438
Mean	MG	–	–	0.838	0.988	0.786	0.824	0.859	0.158	0.800	0.319
46–49	TA	Low	Yes	0.985	0.972	0.944	0.881	0.946	0.583	0.000	0.438
50	TA	Low	No	1.000	1.000	0.875	1.000	0.969	0.667	0.000	0.500
51–54	TA	Medium	Yes	1.000	0.972	0.994	0.914	0.970	0.583	0.500	0.563
55	TA	Medium	No	1.000	0.988	0.996	0.852	0.959	0.333	1.000	0.500
56–59	TA	High	Yes	0.999	0.986	0.985	0.854	0.956	0.667	0.000	0.500
60	TA	High	No	–	–	–	–	–	–	–	–
Mean	TA	–	–	0.996	0.979	0.969	0.889	0.958	0.595	0.214	0.500

medium-level samples, also attempt to draft one or several possible structures for the final answer during the reasoning process.

The low reasoning level is insufficient to help understand the final answer. In contrast, the medium and high levels tend to repeat content extensively without adding meaningful value. This effect is particularly pronounced at the high reasoning level.

Both the medium and high reasoning levels frequently mention data sources such as database or log file names and file paths. While potentially interesting, these details were not requested and are often inaccurate. The model also exhibits difficulties with tool identification, frequently combining the names of well-known tools or assigning them to inappropriate tasks (e.g., Cellebrite Physical Analyzer is cited as an acquisition tool, although this role should be attributed to Cellebrite 4 PC). The mention of a write-blocker is also common during acquisition steps; while required for certain types of storage, it cannot be practically used with smartphones.

The final answers often include numerous elements and detailed substeps that were not explicitly requested. This behavior may stem from the prompt's instruction to provide a ‘detailed methodology’.

4.5.4. Timeline analaysis

The TA represents the most complex task. The input data are sparse and incomplete, and the available event information is limited due to context size constraints. There is no definitive evidence that the file in question was copied to the USB drive or that the Internet connection was activated/deactivated. However, the model must provide a definitive yes-or-no answer and justify its decision, as specified in the prompt. This binary constraint and request for a justification were not purposefully designed to induce a specific behavior, but we observed an interesting model behavior.

In the CoT part, it struggles with the ambiguity between the data and the request (binary decision with justification), often reasoning along the lines of: “We need to answer yes or no. We could say ‘no’ because we cannot confirm that the Internet was deactivated, but we could also say ‘yes’ because there is no evidence of Internet usage.” In the final answer, this binary requirement prevents the model from expressing uncertainty, even though such uncertainty is explicitly discussed during the reasoning process. As a result, certain statements may appear definitive in the final answer while being presented as uncertain in the chain of thought. In some cases, the model even arrives at an incorrect conclusion despite recognizing that the evidence does not clearly support it. In one instance, the model even questioned whether portionsof the timeline might originate from another device.

For this task, access to the reasoning section of the generated text is important, as it reveals the model's difficulty concluding and highlights its awareness of uncertainty.

The model also reflects on user assumptions regarding the Internet being deactivated and the type of answer expected by the user. At higher reasoning levels, it begins to explore unsupported and speculative hypotheses, such as the already mentioned hypothesis that the timeline belongs to another machine, the presence of a local DNS server resolving URLs to local IPs, the use of proxies, or missing data entries.

Therefore, Research Question 4 can be answered as follows: Yes, the reasoning process and the generated CoT can be used to improve the understandability of LLM-generated answers to specific digital forensics tasks, especially in cases where the task includes aspects of ambiguity or if the input data is not complete.

5. Discussion

5.1. Discussion of the results

5.1.1. Quality of the CoT

The results obtained for the chosen metrics indicate good performance for the model's CoT, with an average value of the four metrics systematically over 0.859. In the same way, the qualitative analysis

indicates that many final answers can be better understood/interpreted using the content of the CoTs generated with a medium or high reasoning level.

5.1.2. Quality of the final answer

The quality of the final answer varies significantly, with a lower value for the two tasks that were considered the most complex: the methodology generation and timeline analysis. This seems to indicate that the quality of the final answer is reduced if the input data and request provided in the prompt are complex (timeline analysis) or when the task's success relies on digital forensics common knowledge (methodology generation). Therefore, the final output is likely to be of higher quality for straightforward tasks that depend primarily on the data provided directly in the prompt.

5.1.3. Impact of the reasoning level

Regarding the impact of the reasoning level, the quantitative and qualitative analyses both indicate that the medium reasoning level performs, on average, better. While the reasoning level does not significantly impact the factuality and validity, there is a relation between the reasoning level, the coherence, and the utility. With a higher reasoning level, the model has more 'room' to present the preconditions before making inferences, but it also tends to repeat more content. Therefore, the medium level represents a good balance between coherence and utility. As mentioned previously, there is an exception to the coherence for MG and TA, where the medium level is better than the high level, and an exception for the BHA, where the best CoT score is obtained with the high reasoning level.

5.1.4. Impact of the CoT on the final answer

The quality of the final answer seems to be impacted by the task's difficulty, not by the quality of the CoT. No clear impact of the CoT quality on the final answer quality could be identified. This means that a high-quality CoT could lead to a low-quality final answer, and that a final answer could be of high quality even if the CoT is of low quality.

5.1.5. Utility of the CoT

Based on the obtained results, we consider the model's CoT a useful tool for better understanding and interpreting RLM-generated results. An investigator could, for example, understand why the model determined that an attack took place by examining the reasoning process and reading the explanations, which include more references to logs present in the bash history than the final answer. In scenarios where the request is ambiguous (such as the yes/no request for the timeline analysis), the model shows uncertainty in the CoT while being confident in the final answer. Still, it is insufficient to explain how the model generated the answer (or the CoT) and does not constitute explainable AI. Forgetting this could lead investigators to model over-trust. In such a case, the CoT's positive impact would turn into a factor favoring errors. Therefore, the reasoning process is considered a useful component, but it is not considered sufficient to explain how the model generated the answer. Ultimately, an RLM remains a language model that is trained to predict the most probable next tokens in a given context. It should therefore be regarded as a tool whose statements must always be validated. This is a process in which CoT can be of assistance.

5.2. How could CoT be used?

CoT is how the model "reasons" and its content contributes to the generation of the final answer. While it was not designed to help the user, it may help investigators to understand and verify the obtained results. We believe that the reasoning could be used by investigators and individuals with sufficient technical background. It does not constitute explainable AI, and must therefore be correctly interpreted. When presented to non-technical stakeholders, such as judges or lawyers, investigators must ensure the information from the CoT is correctly understood.

6. Limitations and next steps

The main limitations of this study can be classified into three categories:

Study's Scale: The study is small scaled. It considers a single model, limited in size, not fine-tuned, with a basic prompting system. These choices were made due to the exploratory nature of the experiments, trying to evaluate the potential of the RLMs' CoT. The model's ease of use and balanced size motivated its selection, as it fits the needs of modest computing environments, i.e., a model that is not too small and can fit on many non-professional GPUs. Testing more models of different sizes or fine-tuned could be investigated in future research. This would likely lead to better CoT and final answer quality, as larger models tend to perform better. Studying how more complex prompting systems, such as Retrieval-Augmented Generation (RAG), or variations in sampling temperature, affect the quality of the CoT and the resulting outputs would also be of interest. Moreover, due to the manual nature of the evaluation, only 60 samples (including two that were discarded) distributed over four tasks were evaluated, reducing the generalizability of the results.

Metrics' Choice: While factuality, validity, and utility were easy to understand and evaluate, coherence is a complex metric. Determining if all the preconditions are met before making the inference is difficult, especially in the CoT, with preconditions scattered throughout the previous steps. In future work, more metrics, including automatically computed metrics, could be considered. For example, it might be possible to use a larger LLM as a judge to evaluate our selection of metrics. This would be particularly valuable in a larger-scale study where manual evaluation is not possible.

Evaluation Process: Although metrics were defined to reduce subjectivity during the evaluation process, the three evaluators likely did not evaluate the results in the same way. For example, the evaluators could have divided the CoT into reasoning steps differently, or a reasoning step could be considered normal for one evaluator and incoherent for another. Knowing that it can already vary across evaluators with technical backgrounds, it is likely that the quality measurement/perception of the CoT will also vary when presented to non-technical individuals. Finally, although it helped reduce the impact of the number of steps, the normalization process did not completely remove the bias introduced by the significant size difference between the reasoning processes generated by different levels of reasoning.

7. Conclusion

The limited interpretability of current LLMs reduces their suitability for meeting the requirements of digital forensics. To advance the goal of LLM understandability in digital forensics, we explored the use of the chain of thought (CoT) mechanism implemented in gpt-oss-20b. This model represents a new class of language models specifically optimized for reasoning tasks, designed to generate intermediate reasoning steps before returning the final answer to the user. We selected four common forensic tasks (Suspicious Message Detection, Bash History Analysis, Methodology Generation, and Timeline Analysis) and related prompts. We then provided these prompts to the model, varying the task, the reasoning level, and the generation strategy (sampling activated vs. sampling deactivated). We manually evaluated the reasoning process using four metrics: factuality, validity, coherence, and utility. The final answers were then evaluated based on the correctness and justification provided by the model. Comments from the evaluators were also compiled and presented. This allows us to answer the research question raised in the introduction:

To what extent can the result from the chain of thought mechanism of

reasoning language models help to understand and interpret the obtained results when undertaking digital forensics tasks? We deem the CoT mechanism of the new RLMs as a useful tool providing "supporting details" that help tounderstand, interpret, and validate the models' results. However, CoT alone does not adequately explain how the model arrives at its answer and therefore does not qualify as a form of explainable AI.

Future research should investigate different reasoning models, consider additional forensics tasks, and use a combination of manually and automatically computed metrics to reduce the subjectivity introduced by manual evaluation. It should also evaluate more samples per task to increase the generalizability of the results.

Although the results are encouraging, it is not recommended to use CoT reasoning as an explanation for the final answer, especially if the results are presented in court. CoT should be viewed as providing "supporting details" that help interpret, understand, and verify the final answer.

CRediT authorship contribution statement

Gaëtan Michelet: Conceptualization, Methodology, Software, Investigation, Writing - Original Draft, Visualization. **Janine Schneider:** Methodology, Investigation, Writing - Original Draft. **Aruna Withanage:** Investigation. **Frank Breitinger:** Conceptualization, Writing - Review & Editing, Supervision.

Disclosure of AI-assisted writing tools

Several authors used ChatGPT and Grammarly to help with tasks such as revising, condensing text, and addressing grammatical errors, typos, and awkward phrasing. All AI-generated suggestions were thoroughly reviewed and adjusted where needed to ensure they accurately reflected the authors' intended meaning before being incorporated into the paper.

Declaration of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We would like to thank the reviewers, and especially the shepherd, for their valuable comments and guidance, which greatly improved our paper.

References

Besta, M., Barth, J., Schreiber, E., Kubicek, A., Catarino, A., Gerstenberger, R., Nyczyk, P., Iff, P., Li, Y., Houliston, S., Sternal, T., Copik, M., Kwaśniewski, G., Müller, J., Łukasz, Flis, Eberhard, H., Chen, Z., Niewiadomski, H., Hoefler, T., 2025. Reasoning language models: a blueprint. https://arxiv.org/abs/2501.11223 arXiv:2501.11223.

Cho, S.-H., Kim, D., Kwon, H.-C., Kim, M., 2024. Exploring the potential of large language models for author profiling tasks in digital text forensics. Forensic Sci. Int.: Digit. Invest. 50, 301814. https://doi.org/10.1016/j.fsidi.2024.301814. URL: https://www.sciencedirect.com/science/article/pii/S2666281724001380.

Creswell, A., Shanahan, M., 2022. Faithful reasoning using large language models. https://arxiv.org/abs/2208.14271 arXiv:2208.14271.

Dinis-Oliveira, R.J., Azevedo, R.M.S., 2023. ChatGPT in forensic sciences: a new Pandora's box with advantages and challenges to pay attention. Foren. Sci. Res. 8, 275–279. https://doi.org/10.1093/fsr/owad039.

Hargreaves, C., Breitinger, F., Dowthwaite, L., Webb, H., Scanlon, M., 2024. DFPulse: the 2024 digital forensic practitioner survey. Forensic Sci. Int.: Digit. Invest. 51, 301844. https://doi.org/10.1016/j.fsidi.2024.301844. URL: https://www.sciencedirect.com/science/article/pii/S2666281724001719.

Henseler, H., van Beek, H., 2023. Chatgpt as a Copilot for Investigating Digital Evidence, 3423. CEUR-WS. URL: https://ceur-ws.org/Vol-3423/paper6.pdf.

Huang, J., Chen-Chuan Chang, K., 2023. Towards reasoning in large language models: a survey. In: *Findings of the Association for Computational Linguistics, ACL 2023* Proceedings of the Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics (ACL), pp. 1049–1065. https://doi.org/10.18653/v1/2023.findings-acl.67.

Lang, J.-H., Schreck, T., 2025. Leveraging LLMs for memory forensics: a comparative analysis of malware detection. Dig. Threats. https://doi.org/10.1145/3748263.

Lee, J., Hockenmaier, J., 2025. Evaluating Step-by-step reasoning traces: a survey. https://arxiv.org/abs/2502.12289 arXiv:2502.12289.

Michelet, G., Breitinger, F., 2024. ChatGPT, llama, can you write my report? An experiment on assisted digital forensics reports written using (local) large language models. Forensic Sci. Int.: Digit. Invest. 48, 301683. https://doi.org/10.1016/j.fsidi.2023.301683. https://www.sciencedirect.com/science/article/pii/S2666281723002020.

Michelet, G., Henseler, H., van Beek, H., Scanlon, M., Breitinger, F., 2025. Fine-tuning large language models for digital forensics: case study and general recommendations. Dig. Threats. https://doi.org/10.1145/3748264.

Mondorf, P., Plank, B., 2024. Beyond accuracy: evaluating the reasoning behavior of large language models – a survey. https://arxiv.org/abs/2404.01869 arXiv:2404.01869.

Oh, D.B., Kim, D., Kim, D., Kim, H.K., 2024. volGPT: evaluation on triaging ransomware process in memory forensics with large language model. Forensic Sci. Int.: Digit. Invest. 49, 301756. https://doi.org/10.1016/j.fsidi.2024.301756. URL: https://www.sciencedirect.com/science/article/pii/S2666281724000751.

Patil, A., Jadon, A., 2025. Advancing reasoning in large language models: promising methods and approaches. https://arxiv.org/abs/2502.03671.arXiv:2502.03671.

Plaat, A., Wong, A., Verberne, S., Broekens, J., Van Stein, N., Bäck, T., 2025. Multi-step reasoning with large language models, a survey. ACM Comput. Surv. 58. https://doi.org/10.1145/3774896.

Scanlon, M., Breitinger, F., Hargreaves, C., Hilgert, J.-N., Sheppard, J., 2023a. ChatGPT for digital forensic investigation: the good, the bad, and the unknown. Forensic Sci. Int.: Digit. Invest. 46, 301609. https://doi.org/10.1016/j.fsidi.2023.301609. URL: https://www.sciencedirect.com/science/article/pii/S266628172300121X.

Scanlon, M., Nikkel, B., Geradts, Z., 2023b. Digital forensic investigation in the age of ChatGPT. Forensic Sci. Int.: Digit. Invest. 44, 301543. https://doi.org/10.1016/j.fsidi.2023.301543.

Sharma, B., Ghawaly, J., McCleary, K., Webb, A.M., Baggili, I., 2025. ForensicLLM: a local large language model for digital forensics. Forensic Sci. Int.: Digit. Invest. 52, 301872. https://doi.org/10.1016/j.fsidi.2025.301872. URL: https://www.sciencedirect.com/science/article/pii/S2666281725000113.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q.V., Zhou, D., 2022. Chain-of-thought prompting elicits reasoning in large language models. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (Eds.), Advances in Neural Information Processing Systems, vol. 35. Curran Associates, Inc., pp. 24824–24837. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf

Wickramasekara, A., Breitinger, F., Scanlon, M., 2025a. Exploring the potential of large language models for improving digital forensic investigation efficiency. Forensic Sci. Int.: Digit. Invest. 52, 301859. https://doi.org/10.1016/j.fsidi.2024.301859.

Wickramasekara, A., Densmore, A., Breitinger, F., Studiawan, H., Scanlon, M., 2025b. AutoDFBench: a framework for AI generated digital forensic code and tool testing and evaluation. In: *Proceedings of the Digital Forensics Doctoral Symposium* DFDS '25. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3712716.3712718.

Xu, F., Hao, Q., Zong, Z., Wang, J., Zhang, Y., Wang, J., Lan, X., Gong, J., Ouyang, T., Meng, F., Shao, C., Yan, Y., Yang, Q., Song, Y., Ren, S., Hu, X., Li, Y., Feng, J., Gao, C., Li, Y., 2025. Towards large reasoning models: a survey of reinforced reasoning with large language models. https://arxiv.org/abs/2501.09686.arXiv:2501.09686.