Analyst-In-Loop LLM Systems for Forensic Timeline Analysis Assistance

Author: Oscar J. Rodríguez Ruiz
Advisor: Dr. Jeffrey Duffany
Master in Computer Science
Graduate Project EXPO, February 2026

Abstract

The increasing volume and complexity of digital evidence in digital forensic investigations have made manual timeline analysis, an inefficient and reckless waste of resources. Tools such as Plaso (Log2timeline) have shown to be highly effective at creating "super-timelines" that gather information from various sources. Creating datasets spanning thousands of events for the forensics examiner. However, we have made great strides in the field of artificial intelligence. These allow for the utilization of the processing power provided to assist with the process of detecting anomalies and filtering for essential in an investigation. Through extracting general outputs as CSV files, we can create processes highlighting the role that private AI models will take within the field after considerable training with the available forensic training data.

Introduction

Throughout the years, technology has evolved rapidly, integrating itself into every aspect of our current lives. Forensics has gained a focus around 8 aspects: methods, equipment, mobile, networks, data extraction, crime, identification and the cloud.[1][2]. Technology has thusly begun to accommodate for these changes. One such avenue of investigation has surged with LLMs and how they can become a practical part of our forensic tool suite[3]. This project showcases the ability for LLM based tools to facilitate one such forensic investigation with the understanding of supertimelines through the use of natural language.

Background

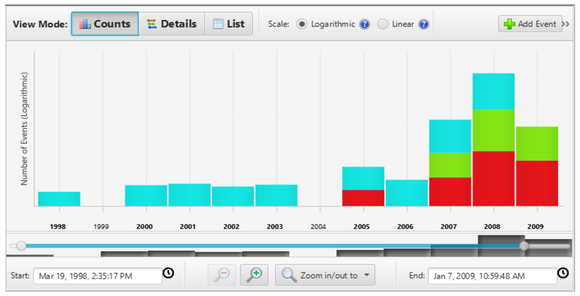
Digital forensics has become a critical discipline that covers important investigative artifacts such as storage media and the cloud. This has led to the increased necessity for event reconstruction in complex investigations. Timeline analysis is a fundamental task for digital forensic examiners as it precisely states what events took place in a system. However, more specialized systems starting in 2009 with Cyber Forensic Time Lab or CTFL by Olsson and Boldt also started gathering metadata from locations such as EXIF data, and Windows Registry [4]. One of these systems is log2timeline, invented by Kristinn Guðjónsson back in 2010 [5]. These have become staples of the forensic examiners arsenal. However these same systems overstimulate our ability to process information and can take days to filter through. This leaves new investigators lost in extensive data pools. Fifteen (15) years ago, it was much more manageable for a forensic investigator to comb through events manually. However, today's technology environment has expanded the amount of data that is more commonly available.

Problem

The challenge we face today is the staggering volume of growth in the amount of data available in all home systems. This "data explosion" has led to higher numbers in storage among all personal digital devices from computers to phones to even your fridge. To address this challenge "supertimelines" UI are the tool utilized by most investigators [6]. While Log2Timeline is exceptional at its job it tends to include a multitude of low-level devices from single disks. New investigators without any context would also require training and better understanding of these tools. Therefore 3 problems arise in the field: human scalability, a semantic gap, and finally a tooling gap which continues to be a challenge faced by most developers and investigators in the field.

Methodology

.This project involved a pipeline designed to ingest forensic data from an E01 file (nps-2009-casper-rw) and output summaries based on that forensic data [7]. The process was divided into data acquisition, pre-processing, and LLM analysis. However, what this project focuses on is creating a read/able timeline assistance file. Autopsy does a good job of showcasing a timeline that you can edit and look through by yourself, but without proper knowledge, you would not be able to understand what it means. Autopsy actually uses Plaso/Log2timeline to create one of these timelines as seen in the figure below however these timelines don't tend to include natural language explanations of the actions taken.



For data acquisition, this project utilizes the same system used by Autopsy but from the command line to process the E01 file using Log2Timeline's psteal.py command to generate a much more parsable csv file. This command allows for the direct creation of a csv file highlighting all the events and actions taken within the image. These create a base for the parsing and understanding power of the LLM. The csv gets chunked into 500 line chunks using the panda library using the code below.

```
df = pd.read_csv(CSV_PATH)
total_rows = len(df)
total_chunks = math.ceil(total_rows / CHUNK_SIZE)

print(f"[+] Loaded CSV with {total_rows} rows")
print(f"[+] Processing in {total_chunks} chunks of {CHUNK_SIZE} rows")
```

The script then takes each chunk of information and utilizing a personally crafted prompt sets rules and criteria which the LLM utilizes in order to objectively analyze the information within the timeline. This prompt is a base of the program however it should be edited and carefully crafted to establish the basis for different investigations. One such criteria prompt can be seen below explaining the criteria for an investigation on an ubuntu system that was downloading files from government websites.

```
CRITERIA = """
- Focus on explicit user-initiated actions
- Prioritize WEBHIST, OLECF, and FILE artifacts
- Emphasize activity in user directories (e.g., /home/ubuntu)
- Ignore baseline system artifacts unless user-accessed
- Ignore entries marked as 'Not a time'
- Highlight file downloads from browsers
"""
```

Results and Discussion

The project experimentation revealed that the system's ability to ingest and filter the forensic timelines and their generated data was astonishing and helpful. What normally would have taken hours of time was summarized with minor effort in anywhere from 1 to 3 minutes per chunk of 500 lines. As well, the original timeline, which was around 60,000 lines of information, was cut down to a tenth of the total amount of lines and set up as natural language, which is easier to understand. The utilization of pandas to reduce the volume of the supertimeline into digestible and important information allowed for a more formal system that mediated token consumption on the API (OpenAI) alongside prompt sizing. The system excelled at maintaining a chronological order across the chunks of log entries fed to it and was able to establish a summarized analysis and understanding of the happenings within each chunk of the CSV, as shown below. This system categorically takes these large files of data and converts them into a bridge for those who are not as technically adept and are still attempting to understand the scope of the information in question.

"=== CHUNK 26 ANALYSIS ===
-**2008-12-28 (21:32:30 to 21:32:55 UTC) – User-Initiated File Downloads via Firefox Browser:**
- Multiple events indicate the user downloaded files from the website hraunfoss.fcc.gov, accessed via www.fcc.gov (not typed directly)
- PDF file "FCC-08-281A5.pdf" downloaded and saved to /home/ubuntu/Desktop/MyStuff/FCC-08-281A5.PDF.
- DOC file "FCC-08-281A6.doc" downloaded and saved to /home/ubuntu/Desktop/MyStuff/FCC-08-281A6.DOC.

The system did excel at handling the large file base, and while it did take some time to fully process the entirety of the images, the 2GB USB image only took around 5 minutes to fully process. However, as the files get larger, the risk of losing context between chunks increases, as well as the risk of hallucinations. These systems seemed reliable and very useful, they were, however, still prone to hallucinations. These summaries created by the AI can be reliable leads in an investigation, but they cannot be treated as absolute truths, as one of the more important parts of maintaining forensic integrity is to maintain provenance of the data.
One part that is lacking is the entirely ethical implementation of the system. A system such as this sends information across the internet to LLMs, which can be catastrophic for the sensitive information withheld in most forensic cases today. Therefore a disclaimer as the one shown below is needed.

DISCLAIMER:
This document contains automated, preliminary analysis generated using a language model.It is provided solely as an analytical aid to assist forensic examiners.It does NOT constitute a forensic finding, expert opinion, or legal conclusion. All interpretations must be reviewed, validated, and corroborated by a qualified forensic examiner.

Conclusions

This project demonstrated the significant potential of leveraging Large Language Models to address the persistent gaps that exist within digital forensic timeline analysis today. By developing with Python as a base and integrating popular LLMs (such as Gemini or OpenAI), this study successfully showcases an automation of the transformation of Disk Images into a natural language format that can facilitate the work of an analyst or student. This also establishes a process for utilizing LLMs as an assistive tool for either studying or practicing supertimeline analysis. While the use of LLMs within this investigation is far from ideal for replacing an expert in the field, they can actively serve as a powerful force multiplier. By reducing cognitive load, these tools may allow users to make higher-level decisions and faster.

Future Work

The current implementation shows a feasible framework for utilizing a Large Language Model to synthesize a simplified version of complex supertimeline outputs. However, there are still areas of investigation that need to be explored and enhanced when it comes to this field, especially when it comes to establishing reliable, scalable systems for deeper analysis. One valid way to remedy this is by exploring techniques for Forensic Data Reduction. Whitelisting things like data telemetry could save on token costs in the long run, and Dynamic Feature selection could be used to optimize the model's focus when looking for specific data. Another valid avenue of study would be to actively compare a system like this to a human in terms of the ROGUE and BLEU scales as presented by Studiawan et al.[8]
The chunk system feeds the entire CSV to the API within multiple prompts, which ultimately takes time and is inefficient if we want to use more natural language within the query. A solution to this system may utilize Retrieval-Augmented Generation, also known as RAG[9].

Acknowledgements

This material is based upon work supported by, or in part by, the National Science Foundation (NSF-SFS) under contract/award 2140638. The author would like to thank Dr. Jeffrey Duffany for his guidance and mentorship throughout this project. Special thanks are also extended to Joann Casillas Tanon for editorial support.

References

[1]Atlam, H. F. (2025). *LLMs in Cyber Security: bridging practice and education. Big Data and Cognitive Computing*, 9(7), 184. https://doi.org/10.3390/bdcc9070184
[2] Jiang, G., & Li, C. (2019). *A Scientometric Review of Research Evolution in Digital Forensics.* Proceedings of the 3rd International Conference on Computer Science and Application Engineering, 1–11. https://doi.org/10.1145/3331453.3362055
[3] Wickramasekara, A., Breitinger, F., & Scanlon, M. (2025). *Exploring the potential of large language models for improving digital forensic investigation efficiency.* Forensic Science International Digital Investigation, 52, 301859. https://doi.org/10.1016/j.fsidi.2024.301859
[4] Olsson, J., & Boldt, M. (2009). *Computer forensic timeline visualization tool.* Digital Investigation, 6, S78–S87. https://doi.org/10.1016/j.diin.2009.06.008
[5] Hargreaves, C., & Patterson, J. (2012). *An automated timeline reconstruction approach for digital forensic investigations.* Digital Investigation, 9, S69–S79. https://doi.org/10.1016/j.diin.2012.05.006
[6] Debinski, M., Breitinger, F., & Mohan, P. (2018). *Timeline2GUI: A Log2Timeline CSV parser and training scenarios.* Digital Investigation, 28, 34–43. https://doi.org/10.1016/j.diin.2018.12.004
[7] Disk Images – Digital Corpora. (n.d.). https://digitalcorpora.org/corpora/disk-images/
[8] Studiawan, H., Breitinger, F., & Scanlon, M. (2025). *Towards a standardized methodology and dataset for evaluating LLM-based digital forensic timeline analysis.* Forensic Science International Digital Investigation, 54, 301982. https://doi.org/10.1016/j.fsidi.2025.301982
[9] Sharma, B., Ghawaly, J., McCleary, K., Webb, A. M., & Baggili, I. (2025). *ForensicLLM: A local large language model for digital forensics.* Forensic Science International Digital Investigation, 52, 301872. https://doi.org/10.1016/j.fsidi.2025.301872