Automated Threat Intelligence Benchmarking with Relevant Fine-Tuned Large Language Models

Andreas Dreeke
Master of Cyber-Security
University of Adelaide
Adelaide, Australia
andreas.dreeke@student.adelaide.edu.au

Prof. Hung Nguyen
Computer Science, SET
University of Adelaide
Adelaide, Australia
hung.nguyen@adelaide.edu.au

***Abstract*—The growing integration of large language models (LLMs) into everyday workflows has opened new opportunities in cybersecurity, particularly for automating threat intelligence and adversarial modelling. This study creates a benchmark for the capabilities of publicly available models to reconstruct incident kill chains from real-world cyber attack reports using the MITRE ATT&CK Flow framework. Each model is evaluated based on confusion matrix metrics for the rubrics of MITRE technique mapping, cyber context understanding, and general flow reconstructions. In addition to determining the best performing model, the result analysis also explores the structure and content of good and bad CTI reports for automated LLM processing.**

I. Introduction

A. Background

Microsoft (Copilot) and Google (Gemini AI) are two prime examples for two global leading technology giants that show a drastic increase to incorporate artificial intelligence (AI) into corporate culture, consumer interaction, and general business models. With the launch of Microsoft Copilot as AI assistant directly integrated into all Microsoft productivity applications, such as Excel, Word, or PowerPoint, a quick solution to repetitive and predictable tasks confronts approximately 350 million paid private users in all age groups, as well as over 1 million companies [1] with the increasing usability of large language models (LLM) to solve everyday tasks. With a very broad user base and no specific customer group, the GPT-4 model developed by OpenAI, which is the foundation for Microsoft's in-app Copilot service [2], is trained in the broad openly accessible knowledge of the Internet without topic prioritisation and therefore equipped to solve any given general objective.

Based on this ”global” training, each publicly available LLM is also familiarised with cybersecurity peculiarities, terminology, and syntax, as a highly specific and niche profession, with ever increasing importance. One of the unique components in the cybersecurity domain is filled by the open-source MITRE ATT&CK[1] database, summarising an up-to-date collection of adversary techniques and procedures in a well-structured form, covering general descriptions, real-life incident observations, best-practice mitigation and detection advices, and thus providing guidance throughout the industry. The well-maintained and detailed distinction between tactics and subtactics allows for distinguishable component evaluation of cybersecurity relevant incidents and a organised representation of the event kill chain offering a simplified way to learn, mitigate, and prevent complex connections in a structured manner. In a more recent approach to equip defence professionals with the understanding of how their adversaries operate as well as the impact on their own organisation, MITRE introduced ATTACK FLOW framework to visualise attack patterns in graphs and share them across the industry. But while use cases affect most cybersecurity job descriptions, from cyber threat intelligence (CTI) collection, the posture of the Blue Team, over threat communication with executives, to incident response, threat hunting, and malware analysis [3], creating a single meaningful attack flow requires sophisticated research, well-founded knowledge, years of experience, and ultimately a huge investment in time.

The described reality of increasing normalisation of LLM usage for general tasks raises the question of the usability of the very same tools for highly specialised tasks in the cybersecurity domain which, until now, required highly trained individuals. With help of automatic analysis of CTI reports through LLM's offer a quick and efficient summary of massive natural text [4]and thus offer quick threat analysis for specific organisations and setups. Furthermore, allows the unification of natural language into well-defined output structures (e.g. JSON) for the possibility to import them into modern SIEM (Security Information and Event Management) systems based on the mapping to the MITRE ATT&CK framework [5]. The same connection to MITRE can also be used to effectively determine the optimal placement of decoys and honeypots and deploy them proactively to enhance overall network security [6].
The scope of this paper is to compare the current state of the largest freely available LLM models to solve complex cybersecurity objectives. Each model is evaluated based on the ability to correctly reproduce incident kill chains as defined by the MITRE ATT&CK framework and ultimatley draw a conclusion on their ability to do so.

[1] https://attack.mitre.org/, Accessed 2025-11-07

B. Goals

This research aims to create a meaningful comparison of publicly accessible and free LLM models, therefore accessible to small organisations and individuals, for complex cybersecurity problems. Conducting the experiments on the example of attack flow reconstructions based on CTI reports integrates the native ability of LLM models to analyse and process natural language (full-text) documents and thus offers a profound ground to benchmark different providers with this task. The result will not only give an insight into the best numerical model based on the calculated metrics, but also provide an overview of the general state of cybersecurity capabilities.

Besides the direct outcome, the created code base allows for an easy adaption of additional and/or updated model versions and can therefore be reused for future research in the same field, allowing quick comparison of upcoming technology and highlighting the progress in research and industry. Furthermore, an existing and fully functional pipeline to convert detailed CTI reports into graphs allows the creation of detailed attack flow reconstructions, providing a time reduction in decision-making processes, vulnerability discovery, and threat mitigation.

C. Challenges

The course of this project required the compromise and shaping of the target in specific directions. Although the preceding project aimed to produce an initial evaluation solely based on the Google Gemini LLM followed by an optimisation process for improved usability and accuracy, the conclusion of the same was, of course, necessary due to a variety of reasons. For one, Google changed its API policy to lock the used Gemini 1.5-pro version behind a paywall, making a switch to a different version or provider necessary. Second, the result metrics showed a clear inadequacy in predicting the order of the kill chain actions, leading to questions regarding the context interpretation capabilities of the specific LLM. And finally, the limited scope of fine-tuning possibilities appropriate for another three-month project led to the decision to adapt the project objective towards a benchmark evaluation of freely accessible LLM models.

The new aim itself is followed by multiple difficulties, as the acquisition of API access to the largest and publicly known LLM providers is mostly inaccessible to free users, which was defined as a direct goal when shaping the project goal. This leads to relying on older versions for some providers, compared to free access to other models with a limit in parameters such as send/received tokens per minute and request per day. Unlike the first impression of unfair competition due to not always implementing the latest version, a closer look revealed the possibility for a more profound recommendation as the model accessibility is also a criterion when evaluating the usability in regards of small-scale usage.

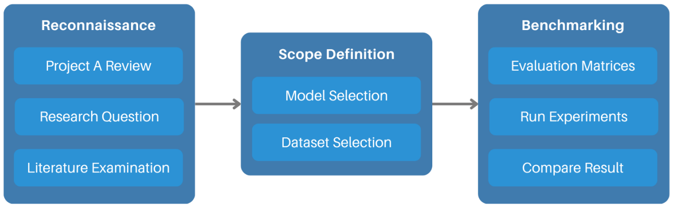

Fig. 1. Three phased project process.

Besides usual project hurdles, a last obstacle worth mentioning is the inconsistency of the generated output, which already became visible in the preceding work. The same model, with the same query and input, generated different results when tasked repeatedly, which highly challenges the need for scientific reproducibility and honesty. Mitigation and production of a reliable result is achieved by conducting and evaluating every experiment multiple times, followed by a combined averaging to create a final result. This leads to a meaningful impression and a low chance of inconsistencies and randomness in the result analysis.

II. Methodology

This project follows an experimental research design with all code written in Python, using Jupyter Notebook files in Google Colab, and version controlled with GitHub. The procedure of this project is divided into three major components (Fig. 1). Before all, the research question of creating a comparison of different and free LLM models was defined on the example of a complex and context-dependent cybersecurity problem. This principle guided the search for related literature for the cybersecurity domain in regards of other benchmarking analysis, general AI and the subtopic of large language models, and best practices in prompt definitions to achieve the best possible outcome.

Following the reconnaissance phase, the scope for test models and ground-truth comparison is defined. The list of compared LLM providers consists of known industry competitors and compares the usability of: DeepSeek, OpenAI, Meta, Google, and MistralAI. The ground truth to test each experiment is chosen as the MITRE ATTACK FLOW database consisting, at this point (July 2025), of 38 verified and professionally modelled kill chains with referenced resources used for the creation of each, giving the chance to fully reproduce the already existing graph. In each experiment iteration, a model is tasked with creating a detailed attack reconstruction based on the same data, and then compared to the ground truth defined by the varying MITRE authors.

Lastly, a framework is created to evaluate each generated output based on the same criteria, allowing a fair and objective benchmark comparison. For an automated evaluation process of 20 graphs (4 models * 5 experiments) per incident, the provided JSON representations are compared to the

TABLE I
CONFUSION MATRIX COMPONENT DEFINITION FOR RESULT EVALUATION

True Positive	Exists in ground truth and experiment
False Positive	Exists only in experiment
False Negative	Exists only in ground truth
True Negative	Exists neither in ground truth nor experiment

generated JSON output of each experiment and evaluated based on identified attack techniques, affected assets, and flow consistency. Table I shows the definitions of the confusion matrix that are used to finalise the evaluation metrics. As true negatives (TN) are defined as not existing in both considered graphs, they cannot be captured by any means. Therefore, all comparison metrics focus on TN independent calculations except accuracy, where TN is always set to zero.

The following evaluation metrics are selected based on their significance with respect to different criteria. With the limitations of non-existent true negatives, the calculated accuracy for each model describes the proportion of correctly identified objects from the ground truth. An accuracy of 100 % is equal to a complete match between the generated graph and the MITRE graph.

$$\text{Accuracy} = \frac{TP}{TP + FP + FN} \tag{1}$$

The precision metric describes the relation of correctly and wrongfully predicted objects to measure the false positive rate. In the context of MITRE ATT&CK reconstuctions this metric visualises the prediction effectivness of each model, as a low precision score is equivalent to random guesses and results in overly full graphs reducing their value as quick and informative structures.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

The recall rate evaluates how often an actual positive value is missed, and is in this context therefore just as important and meaningfull as the precision. Missed steps might result in unresolved vulnerabilities for the reader of the graph and could result in security risks.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

Lastly, the F1-Score is used to ultimately benchmark all models and determine the top performer. As the precision- and recall rate are equally important, the F1 score provides a solid trade-off to evaluate the general performance of each model.

$$\text{F1 Score} = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{4}$$

III. LITERATURE REVIEW

A. Large Language Models

Large language models (LLM) are defined as transformer models summarising billions of parameters, hence ”large”, that are trained on an immense amount of textual data [7].

TABLE II
GENERAL SEPARATION OF ARTIFICIAL INTELLIGENCE SUB-CATEGORIES.

Category	**Definition**	**Subcategory**
Machine Learning	Use training data to conduct predictions	Supervised Unsupervised Reinforcement Deep Learning
Natural Language Processing	Interpret and generate human language	Text Classification Large Language Models Named Entity Recognition
Computer Vision	Interpret and process visual input	Image Classification Object Detection

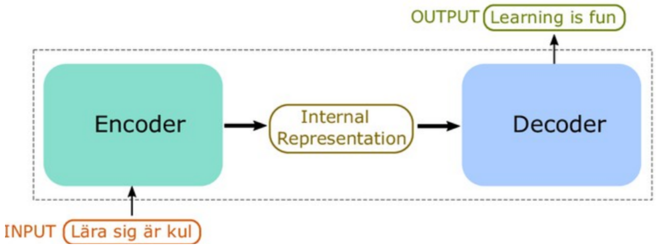

Fig. 2. Simplified transformer composition with foreign language as input in the encoder, and english language as output of the decoder [10].

For a general understanding of large language models, it is helpful to first analyse the location within the large network of AI subdomains visualised in Table II. The usage of LLMs in Human-Machine interaction (e.g. ChatGPT) through full text sentences and human language (= Natural Language [8]), they get classified in the category of natural language processing (NLP), but the capabilities to solve complex tasks is obtained through excessive prior training, where state of the art models (e.g. LLaMa) train on at least 5 Terrabytes of online available data, consisting of sources like Github, Wikipedia, Books, and a general collection of accessible web pages collected with web crawlers [9]. In general, todays definition of a large language model is a combination of transformer-based deep learning process in combination with natural language processing.

Compared to conventional machine learning techniques (e.g. neural networks), transformer-based models do not rely on stacking multiple abstraction layers. Instead they consist of an encoder, decoder, and an attention module (Fig. 2). The encoder is used to convert natural language into tokens (= word fractions) as internal representation, while the decoder is reverting the process by taking internal representations to form the desired output (e.g. translation to different language) . The attention module takes the role of the stacked network layers in e.g. convolutional neural networks (CNN) or recurrent neural networks (RNN) by analysing the relationships within a sequence of tokenised input and gradually learn the embodied pattern and interactions and finally apply a weight (= model parameter) to the learnt dependency [10].

TABLE III
Comparison of previously conducted benchmarking with content of this paper

Author	DeepSeek	Meta AI	Google	OpenAI	Mistral	Real-World	MITRE
D. Ristea et al. [11]	-	-	X	X	-	-	-
Y. Zhu et. al [12]	-	-	-	-	-	X	-
M. Kouremetis et al. [13]	X	X	-	-	X	-	X
A. Dawson et al. [14]	X	X	X	X	-	-	-
Our work	X	X	X	X	X	X	X

B. Related Work

The use of artificial intelligence in combination with cybersecurity, while still developing in fields outside of big data analysis in e.g. security information and event management (SIEM) systems, is not a new approach. Experimental approaches like Microsoft's CyberBattleSim research capabilities of offensive and defensive machine learning models trained in reward-based reinforcement techniques in a mock capture of the flag (CTF) environment [15] and make use of the progress in AI research and hardware component capabilities.

Closer to the topic of this paper is the work of Dan Ristea et al. investigating the LLM capabilities of different models of OpenAI, Anthropic and Google to exploit given vulnerabilities in a system with OpenAI's o1 model (2024) that is capable of successfully exploiting 67% of the vulnerabilities assigned [11]. Other benchmarking evaluations focused on challenging the results of previous benchmarking of mocked CTF environments (e.g. CyberBattleSim) by using the same models to exploit real-world services, which lead to a drastic decrease in the success rate from 60% to 13% [12]. In a different experiment, M. Kouremetis et al. evaluated the ability of models like LLaMa, DeepSeek, or Mistral to correctly identify MITRE attack techniques based on incident description and reached a prediction accuracy of up to 90% for their best model [13]. However, each LLM is given a multiple-choice setup to correctly identify the correct MITRE id, which already requires preparatory work and does not challenge each model with raw data.

Compared to previous benchmark research on the use of LLM models for complex cybersecurity tasks, only Y. Zhu et al. [12] included real-world scenarios in the evaluation process, but did not compare the results for a broad spectrum of openly available models. Opposed to that, D. Ristea et al. [11], M. Kouremetis et al. [13], and A. Dawson et al. [14] implemented a range of different models, but lacked the jump out of laboratory conditions and mock scenarios. The content of this paper summarises the effort to combine both into a combined research on the effectiveness of LLM problem solving skills on complex cyber security tasks.

IV. Experimental Setup

A. Benchmark Pipeline

Although the goal of the project changed, the initial pipeline process is maintained and the benchmarking process extends the code base established in previous project A. The subordinate steps, compared in Table IV, show a high similarity and reusability of previously generated code that require closer investigation to highlight the additional work carried out, especially in relation to steps two and four, while step 3 required only minor adaption by removing unnecessary legacy code.

TABLE IV
Macro pipeline comparison of project A & B.

#	Project A	Project B
1.	Read CTI	Read CTI
2.	Send LLM Query	Send LLM Query
3.	Generate Graph	Save JSON response
4.	Evaluate	Evaluate

1) LLM Query: The first project part focused solely on the CTI processing and evaluation of the Google Gemini model and, therefore, required only a single API class. Part B, on the other hand, aims to compare the capabilities of the previous task based on additional models, and thus create a benchmark comparison between each. To blend in with the existing code base and allow errorless integration of the additional models, each API integration follows the subsequent layout.

```
class LlmAPI:
    #Define Parameters
    def __init__(self):
      ...

    #API Call
    def get_response(self, query):
      ...

    #API Call Wrapper
    def get_full_response(self, query, images=[]):
      ...
```

Listing 1. Generic LLM API class.

The implemented API classes include versions for Google Gemini, OpenAI GPT-4o, Meta AI LLaMa 3, DeepSeek-v3 and Mistrals ministral-8b-2410. A complete comparison and a detailed listing of each model in terms of free usability are given in Table V. Toward the middle of this paper Google published an openly accesible version of their Gemini 2.5 pro model, but initial test runs revealed connectivity problems resulting in failed and incomplete interactions, due to high network utilisation on their free-access endpoint. Thus, the benchmarking process continued with 2.5 Flash instead.

2) Evaluation: To automate the evaluation of the ground truth graph and the experimental graph, the JSON representation of each is loaded into the object representations. Listing 2 shows an example for each type of entry in the MITRE database and the possible data that can be extracted.

```
{
```

TABLE V
LLM SPECIFICATION PER MODEL USED.

Developer	Version	API Provider	Requests/Day	Token/Request	Multimodality	Parameter
Google	Gemini 2.5 Flash	Google [a]	250	25,000	Yes	Not disclosed
OpenAI	GPT-4o	ChatAnywhere [b]	5	4,096	Not API Supported	200 billion
DeepSeek	DeepSeek-v3	ChatAnywhere [b]	30	4,096	Text only	671 billion
Meta AI	LLaMa 3-70B	Groq [c]	14,400	6,000	Text only	70 billion
Mistral AI	ministral-8b-2410	Mistral AI [d]	Dynamic	128,000	Text only	8 billion

[a] https://ai.google.dev/gemini-api/docs/models; 16.07.2025
[b] https://github.com/chatanywhere/GPT_API_free; 16.07.2025
[c] https://console.groq.com/home; 16.07.2025
[d] https://mistral.ai/; 16.08.2025

```
    "type": "attack-action",
    "id": "attack-action--0c739735-a984-4897-bc2f-a8737deff66f",
        ...
    "technique_id": "T1195.002",
        ...
    "effect_refs": [
      "attack-action--b9ac316b-c623-4fe3-8ab2-eb8673081edc"
    ]
},
{
    "type": "tool",
    "id": "tool--a4b72723-7fd9-45d1-a108-1559637aa9eb",
        ...
    ]
},
{
  "type": "relationship",
  "id": "relationship--7ffadf55-4741-4af4-8482-4e67328cf6a7",
        ...
  "source_ref": "attack-action--f69ad6fa-05c5-48b9-bfea-7201321c6909",
  "target_ref": "tool--a4b72723-7fd9-45d1-a108-1559637aa9eb"
},
```

Listing 2. Example entry types in MITRE ATTACK FLOW framework.

Each entry in both JSON files is read and initialised in the respective class representation shown in Listing 2, allowing an evaluation based on self-defined criteria to find matches (= True Positives) with a varying level of difficulty to implement.

```
#Representation of each action in the graph
class Action:
  def __init__(self,name, unique_id, tactic_id, technique_id,description,connection=None):
     self.name = name
     self.unique_id = unique_id
     self.tactic_id = tactic_id
     self.technique_id = technique_id
     self.description = description
     self.assets = []
     self.children = connection if connection is not None else []
  def __eq__(self, other):
        if isinstance(other, Action):
            return self.technique_id == other.technique_id
        return False
    ...
# Representation of each asset in the graph
class Asset:
  def __init__(self, asset_type, name, unqiue_id, description):
    self.asset_type = asset_type
    self.name = name
    self.unqiue_id = unqiue_id
    self.description = description
    ...
#Representation of an operator in the graph
class Operator:
  def __init__(self, operator,connection):
    self.operator = operator
    self.children = connection
    ...
```

Listing 3. Generic LLM API class.

a) True Positive Actions: To qualify as a true predicted kill chain step, a technique id predicted in the experiment must be matched to the same technique id at any place in the ground truth. The ability to correctly predict true positive actions directly measures each model capability to match keyword embedded in the natural language CTIs to the often generalised technique descriptions of MITRE. Since MITRE techniques are frequently used in multiple cyber defensive tools, an accurate action prediction is essential for real-world applications.

b) True Positive Asset: As visible in Listing 2, assets (e.g. ”type”=”tool”) are not connected to specific ids but are only equipped with full-text descriptions and title, as well as the connection to the corresponding action. Although the amount of assets per action in ground truth and experiment could be compared, it would contain no meaning in regards to correctness of the asset. A true positive asset is therefore determined by comparing definitions for the list of assets in the ground truth with the list of assets in the experiments in an LLM query, where as each element in the reference data can only be matched once to rule out and penalise duplicates. As an independent and unbiased referee serves Moonshot AI[2] model Kimi-K2.
The asset prediction is a tool to directly evaluate each model's capability to understand enabling and supportive pieces in the attack chain and, therefore, directly measure the models context understanding in regards of complex cyber security problems.

c) True Positive Relationship: A correct prediction can only be achieved for actions that are correctly predicted. For correct actions, check all outgoing connections to the next action(s) and connected assets. If a pair of parent-child actions

[2] https://www.moonshot.ai/

or an action-asset connection is correctly identified, a true positive is counted. Evaluating the number of correctly identified relationships of actions and assets is a direct indicator of each model's ability to correctly recreate the attack flow, as a relationship precision of 100% would equal a complete match of the MITRE attack flow and the predicted reconstruction based on the CTI reports. The relationship metric is directly related to the results of the metric and asset evaluation, as zero matched actions will automatically result in zero predicted relationships.

B. Prompt Optimisation

A proper LLM query, especially in the context of a benchmarking process, is required to follow a defined structure to produce comparable results. An agreed structure for a well-defined prompt generally consists of the following parts: Instructions, Input Data, Context, and Output Indicator. [16].

1) Instructions and Output Indicator: The goal of this benchmarking process, as defined previously, is to automatically evaluate the performance of each model based on the same criteria with the option of easily adapting to additional models. This allows for no margin in the design of the instructions and the output indicator. For a consistent LLM output across all providers and models, the instruction is defined below.

```
Process the given section of a Cyber Threat
Intelligence (CTI) report and identify all attacker
actions associated with MITRE ATT&CK techniques to
recreate the attack flow
```

Listing 4. Instruction section of LLM benchmarking prompt.

```
Your response should be structured as follows (
    sample JSON for guidance):
{{techniques[
    {{
      "action_name": "Example technique",
      "tactic_id": "TA000X",
      "technique_id": "TXXXX",
      "label":["action_name"-"technique_id"],
      "affected_assets": ["Example Affected Asset
    1","Example Affected Asset 2"],
      "prerequisite": ["Action Name - Technqiue ID
    Prior Action in the flow"]
    }}]
}}
If you use double quotes (\") in any results, please
     escape them with \ to avoid poor JSON formating
    . Respond with only the required JSON, DO NOT
    include any preamble or other comments, return
    only the JSON.
```

Listing 5. Output section of LLM benchmarking prompt.

2) Context: The amount of context given along with the instructions directly influences the quality of the solved task [16]. To optimise the query context, this section centres on a series of experiments to compare the generated output of different levels of context depth on the text-based LLama 3-70B model. Table VI shows the definition of each context section, where no content is given for the lowest detail level, while plenty of redundancies and detailed requirements are provided for the most detailed composition.

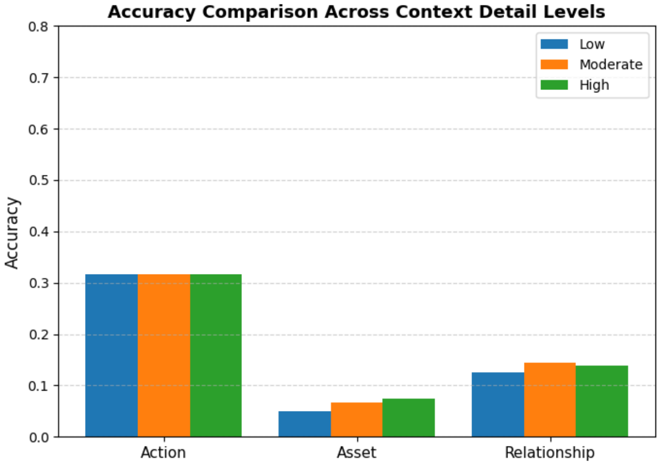

Fig. 3. Accuracy comparison for different levels of context detail for the prediction of actions, assets, and relationships.

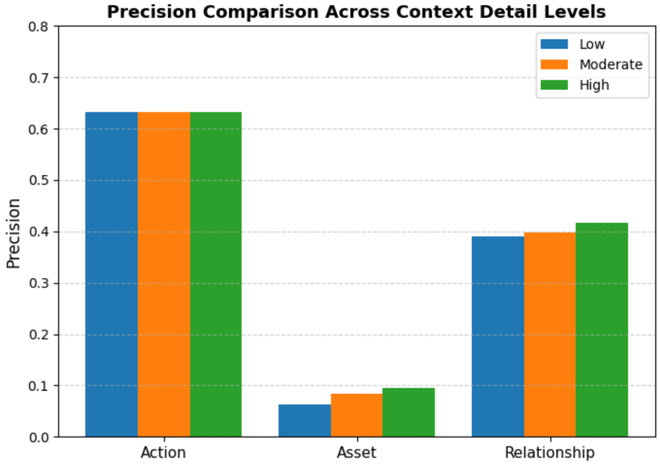

Fig. 4. Precision comparison for different levels of context detail for the prediction of actions, assets, and relationships.

Each query is used five times with the averaged results for the confusion matrix metrics of true positives (TP), false positives (FP), false negatives (FN), and the corresponding results of precision and precision for the lowest level of context detail (Table VII), intermediate (Table VIII) and highest (Table IX).

The graphical visualisation in Fig. 3 shows the proportion distribution of all predictions, while Fig. 4 highlights the proportion of correctly predicted positives. Both graphs show similar results for the experiment, where different levels of context detail did not influence the predicted existence of MITRE techniques (actions) used, but the predictions for assets and correctly identified relationship show an improved outcome with increased context detail.

3) Input Data: Although the general outline of the input data is determined by the prompt instructions as ”section of a CTI report”, the scope of that section is undefined and variable. To determine the impact of varying input depth,

TABLE VI
DIFFERENT LEVEL OF CONTEXT DETAIL FOR THE LLM QUERY.

Detail Level	Context
Low	N/A
Moderate	Identify attacker actions in the order they occurred, along with affected assets and any prerequisites between actions. When assigning MITRE techniques, be as precise as possible based only on the report's content. Output your findings in JSON format with: Action Name Tactic ID Technique ID/Sub-technique ID Label(s) Affected Assets Prerequisites
High	...from the viewpoint of a cybersecurity analyst with significant expertise in the MITRE ATT&CK framework. In order to achieve this, you should do the following: * Analyze the text sentence-by-sentence, if and when necessary, consider additional sentences for context, to identify specific actions taken by attackers. * Consider only the report as source before creating your response * Work out the initial attack, which was used to gain access to the system and made the subsequent attacks viable. * Include each and every step the attackers take, any and all steps, both the obvious and subtle. * Hierarchy of prerequisite attacks should be mostly linear, as these are steps taken in order, with occasional branching and rejoining. * Ensure compilation of all the assets compromised in the attack, even the most minor asset compromised should be noted. * Be as detailed as possible when listing all techniques, if an asset is compromised, the system compromised to achieve this should be noted. * Any specific script/program/malware used must be noted and added to assets. For each action/step identified, you will output information in JSON format with the following structure: 1. Action Name: The specific technique utlised/action taken by the attacker, as described in the text, like 'Vulnerability Scanning', 'Exploit Public-Facing Application', etc. 2. Tactic ID: The Tactic ID from the MITRE ATT&CK framework that categorizes the overarching goal of the action (e.g., TA0001 for Initial Access). 3. Technique ID/Sub-technique ID: The specific Technique or Sub-technique ID from the MITRE ATT&CK framework that the action corresponds to. 4. Label(s): the technique concatenated with the Technique id, with a singular hyphen between, returned as an array with a single element (eg. ["Spearphishing Attachment - T1566"]) 5. Affected Assets: The asset(s) targeted or compromised by the action, based on the report's context. 6. Prerequisite: the action name and technqiue id of any technique that must be completed previously in the attack for this step to take place. If there are no prerequisites, leave the array empty

TABLE VII
AVERAGE RESULT FOR THE EVALUATION OF ALL FILES WITH A LOW LEVEL OF CONTEXT DETAIL.

Type	TP	FP	FN	Accuracy	Precision
Action	12	7	19	0.316	0.632
Asset	1.6	17.4	5.8	0.066	0.084
Relationship	6.4	10	28	0.144	0.397

TABLE VIII
AVERAGE RESULT FOR THE EVALUATION OF ALL FILES WITH A MODERATE LEVEL OF CONTEXT DETAIL.

Type	TP	FP	FN	Accuracy	Precision
Action	12	7	19	0.316	0.632
Asset	1.6	17.4	5.8	0.066	0.084
Relationship	6.4	10	28	0.144	0.397

TABLE IX
AVERAGE RESULT FOR THE EVALUATION OF ALL FILES WITH A HIGH LEVEL OF CONTEXT DETAIL.

Type	TP	FP	FN	Accuracy	Precision
Action	12	7	19	0.316	0.632
Asset	1.8	17.2	6.8	0.074	0.095
Relationship	5.8	7.8	28	0.138	0.417

especially with the varying capabilities of each model as shown in Table V, exemplary experiments are carried out on the Google Gemini 1.5 pro model. Compared to the previous numerical evaluation based on accuracy and precision metrics, this section focuses on the output structure and the generation of realistic graphs with relevant and accurate content.

a) Baseline: The initial setup to serve as comparison for later experiments is done by using general, online available data collected by a preceding project. The result is a five-segmented graph with accurate actions but little detail compared to the original MITRE ATT&CK FLOW representation, with a massive lack of information depth. Furthermore, the described actions, while containing true information, are quite vague and do not meet the expectation of cyber-security professionals.

b) MITRE References: After the initial graph creation, the input is now replaced by an incident report with high technical detail[3] referenced in the offical MITRE attack flow, and therefore a foundation for the creation of the ground truth used to evaluate the outcome. is given to the LLM-model with the query to generate a detailed replication of the incident procedure. The result is a flow split into nine actions and thus almost twice as detailed as the first experiment. The graph shown in Fig. 5. develops first resemblance with the MITRE diagram. The first three segments accurately describe the supply chain attack on Orions SolarWinds, the infiltration of the software development process, the injection of arbitrary code, and the deployment of the injected backdoor through a software update. Additionally, the action assets such as the affected "SolarWinds.Orion.Core.BusinessLayer.dll" are correctly identified. Although greatly improved, it is still quite generic compared to the very detailed incident recreation by MITRE, consisting of roughly 30 actions.

c) Multiple MITRE References: Based on the successful improvement of the output graph, the next experiment targets the question whether or not the results improve when the LLM-model is provided with all references linked by MITRE. The result shown in Fig. 6 appears quite confusing at first glance. The model generated two separate chains, one of them even having circular dependencies. Although this initially appears like a step back, a more detailed look revealed an on-point description for the existing segments showing an

[3] https://www.microsoft.com/en-us/security/blog/2020/12/18/analysing-solorigate-the-compromised-dll-file-that-started-a-sophisticated-cyberattack-and-how-microsoft-defender-helps-protect/; Accessed: 25.07.2025

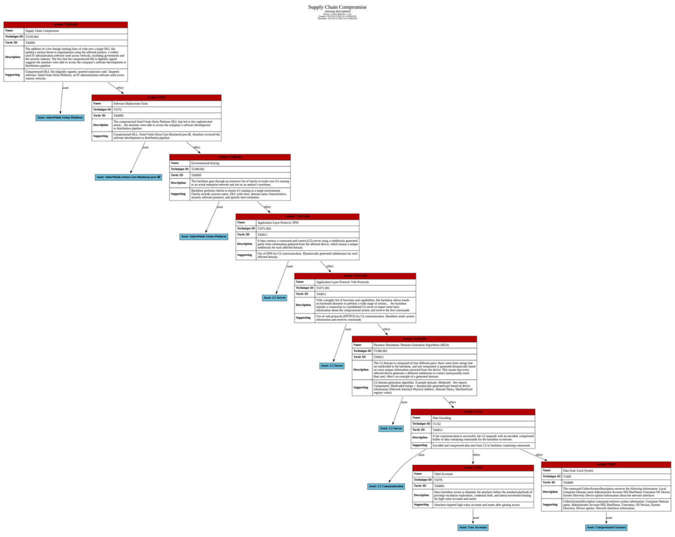
Supply Chain Compromise
(missing description)
Action: T1195.001
Name
Supply Chain Compromise
Technique ID
T1195.001
Tactic ID
TA0001
Description
The addition of a few benign-looking lines of code into a single DLL file spelled a serious threat to organizations using the affected product, a widely used IT administration software used across verticals, including government and the security industry. The fact that the compromised file is digitally signed suggests the attackers were able to access the company's software development or distribution pipeline.
Supporting
Compromised DLL file (digitally signed), inserted malicious code. Targeted software: SolarWinds Orion Platform, an IT administration software used across various verticals.
asset
effect
Asset: SolarWinds Orion Platform
Action: T1072
Name
Software Deployment Tools
Technique ID
T1072
Tactic ID
TA0005
Description
The compromised SolarWinds Orion Platform DLL that led to this sophisticated attack... the attackers were able to access the company's software development or distribution pipeline.
Supporting
Compromised DLL: SolarWinds.Orion.Core.BusinessLayer.dll. Attackers accessed the software development or distribution pipeline.
Asset: SolarWinds.Orion.Core.BusinessLayer.dll
Action: T1480.001
Name
Environmental Keying
Technique ID
T1480.001
Tactic ID
TA0005
Description
The backdoor goes through an extensive list of checks to make sure it's running in an actual enterprise network and not on an analyst's machines.
Supporting
Backdoor performs checks to ensure it's running in a target environment. Checks include: process name, DLL write time, domain name characteristics, security software presence, and specific host resolution.
Asset: SolarWinds Orion Platform
Action: T1071.004
Name
Application Layer Protocol: DNS
Technique ID
T1071.004
Tactic ID
TA0011
Description
It then contacts a command-and-control (C2) server using a subdomain generated partly from information gathered from the affected device, which means a unique subdomain for each affected domain.
Supporting
Use of DNS for C2 communication. Dynamically generated subdomains for each affected domain.
Asset: C2 Server
Action: T1071.001
Name
Application Layer Protocol: Web Protocols
Technique ID
T1071.001
Tactic ID
TA0011
Description
With a lengthy list of functions and capabilities, this backdoor allows hands-on-keyboard attackers to perform a wide range of actions... the backdoor initiates a connection to a predefined C2 server to report some basic information about the compromised system and receive the first commands.
Supporting
Use of web protocols (HTTP/S) for C2 communication. Backdoor sends system information and receives commands.
Asset: C2 Server
Action: T1568.002
Name
Dynamic Resolution: Domain Generation Algorithms (DGA)
Technique ID
T1568.002
Tactic ID
TA0011
Description
The C2 domain is composed of four different parts: three come from strings that are hardcoded in the backdoor, and one component is generated dynamically based on some unique information extracted from the device. This means that every affected device generates a different subdomain to contact (and possibly more than one). Here's an example of a generated domain:
Supporting
C2 domain generation algorithm. Example domain: (Redacted - See report). Components: Hardcoded strings + dynamically generated part based on device information (Network Interface Physical Address, Domain Name, MachineGuid registry value).
Asset: C2 Server
Action: T1132
Name
Data Encoding
Technique ID
T1132
Tactic ID
TA0011
Description
If the communication is successful, the C2 responds with an encoded, compressed buffer of data containing commands for the backdoor to execute.
Supporting
Encoded and compressed data sent from C2 to backdoor containing commands.
Asset: C2 Communication
Action: T1078
Name
Valid Accounts
Technique ID
T1078
Tactic ID
TA0003
Description
Once backdoor access is obtained, the attackers follow the standard playbook of privilege escalation exploration, credential theft, and lateral movement hunting for high-value accounts and assets.
Supporting
Attackers targeted high-value accounts and assets after gaining access.
Asset: User Accounts
Action: T1005
Name
Data from Local System
Technique ID
T1005
Tactic ID
TA0009
Description
The command CollectSystemDescription retrieves the following information: Local Computer Domain name Administrator Account SID HostName Username OS Version System Directory Device uptime Information about the network interfaces
Supporting
CollectSystemDescription command retrieves system information: Computer Domain name, Administrator Account SID, HostName, Username, OS Version, System Directory, Device uptime, Network interfaces information.
Asset: Compromised Systems

Fig. 5. Attack flow diagram of SolarWinds incident with original MITRE reference

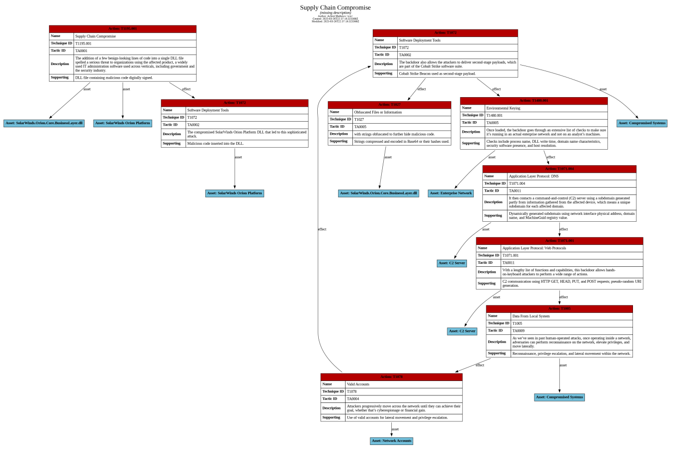
Supply Chain Compromise
(missing description)
Author: Achint Mathews <n/a>
Created: 2025-03-26T22:37:16.323368Z
Modified: 2025-03-26T22:37:16.323368Z
Action: T1195.001
Name
Supply Chain Compromise
Technique ID
T1195.001
Tactic ID
TA0001
Description
The addition of a few benign-looking lines of code into a single DLL file spelled a serious threat to organizations using the affected product, a widely used IT administration software used across verticals, including government and the security industry.
Supporting
DLL file containing malicious code digitally signed.
asset
effect
Asset: SolarWinds.Orion.Core.BusinessLayer.dll
Asset: SolarWinds Orion Platform
Action: T1072
Name
Software Deployment Tools
Technique ID
T1072
Tactic ID
TA0002
Description
The compromised SolarWinds Orion Platform DLL that led to this sophisticated attack.
Supporting
Malicious code inserted into the DLL.
Asset: SolarWinds Orion Platform
Action: T1072
Name
Software Deployment Tools
Technique ID
T1072
Tactic ID
TA0002
Description
The backdoor also allows the attackers to deliver second-stage payloads, which are part of the Cobalt Strike software suite.
Supporting
Cobalt Strike Beacon used as second-stage payload.
Asset: Compromised Systems
Action: T1027
Name
Obfuscated Files or Information
Technique ID
T1027
Tactic ID
TA0005
Description
with strings obfuscated to further hide malicious code.
Supporting
Strings compressed and encoded in Base64 or their hashes used.
Asset: SolarWinds.Orion.Core.BusinessLayer.dll
Action: T1480.001
Name
Environmental Keying
Technique ID
T1480.001
Tactic ID
TA0005
Description
Once loaded, the backdoor goes through an extensive list of checks to make sure it's running in an actual enterprise network and not on an analyst's machines.
Supporting
Checks include process name, DLL write time, domain name characteristics, security software presence, and host resolution.
Asset: Enterprise Network
Action: T1071.004
Name
Application Layer Protocol: DNS
Technique ID
T1071.004
Tactic ID
TA0011
Description
It then contacts a command-and-control (C2) server using a subdomain generated partly from information gathered from the affected device, which means a unique subdomain for each affected domain.
Supporting
Dynamically generated subdomain using network interface physical address, domain name, and MachineGuid registry value.
Asset: C2 Server
Action: T1071.001
Name
Application Layer Protocol: Web Protocols
Technique ID
T1071.001
Tactic ID
TA0011
Description
With a lengthy list of functions and capabilities, this backdoor allows hands-on-keyboard attackers to perform a wide range of actions.
Supporting
C2 communication using HTTP GET, HEAD, PUT, and POST requests; pseudo-random URI generation.
Asset: C2 Server
Action: T1005
Name
Data From Local System
Technique ID
T1005
Tactic ID
TA0009
Description
As we've seen in past human-operated attacks, once operating inside a network, adversaries can perform reconnaissance on the network, elevate privileges, and move laterally.
Supporting
Reconnaissance, privilege escalation, and lateral movement within the network.
Asset: Compromised Systems
Action: T1078
Name
Valid Accounts
Technique ID
T1078
Tactic ID
TA0004
Description
Attackers progressively move across the network until they can achieve their goal, whether that's cyberespionage or financial gain.
Supporting
Use of valid accounts for lateral movement and privilege escalation.
Asset: Network Accounts

Fig. 6. Attack flow diagram of SolarWinds incident with multiple MITRE references

improvement to the previous outcome. The amount of text with lots of unnecessary information is reduced to concise sentences for each segment containing a spit of truth that can be found in the original by MITRE as well. The form appears to be a result of duplicate information existing in each handed document, respectively, but described with different taxonomy and a different focus in the following steps of the incident descriptions. Therefore, the result of the third experiment can also be rated a success, as it further improves the description of the individual action, even if the chain flow itself gets interrupted.

d) Multimodality: The final experiment aims to make use of the capability of some LLM models to process and interpret diagrams and images that are extracted from the CTI reports. Using the same documents as in the third conducted experiment, all changes in this experiment can be traced back to the addition of image processing. The attack flow shown in Fig. 7 is the result of the analysis conducted with a detailed analysis of each step, as described in the following summary:

1) ”The addition of a few benign-looking lines of code into a single DLL file spelled a serious threat to organizations using the affected product, a widely used IT administration software used across verticals, including government and the security industry.”
2) ”The fact that the compromised file is digitally signed suggests the attackers were able to access the company's software development or distribution pipeline.”
3) ”As a result, the DLL containing the malicious code is also digitally signed, which enhances its ability to run privileged actions—and keep a low profile.”
4) ”Once loaded, the backdoor goes through an extensive list of checks to make sure it's running in an actual enterprise network and not on an analyst's machines.
5) ”It then contacts a command-and-control (C2) server using a subdomain generated partly from information gathered from the affected device, which means a unique subdomain for each affected domain.”
6) ”It then contacts a command-and-control (C2) server using a subdomain generated partly from information gathered from the affected device”
7) ”With a lengthy list of functions and capabilities, this backdoor allows hands-on-keyboard attackers to perform a wide range of actions.”
8) ”If the communication is successful, the C2 responds with an encoded, compressed buffer of data containing commands for the backdoor to execute.”
9) ”As we've seen in past human-operated attacks, once operating inside a network, adversaries can perform reconnaissance on the network, elevate privileges, and move laterally.”
10) ”The backdoor also allows the attackers to deliver second-stage payloads, which are part of the Cobalt Strike software suite.”

With only a duplicated message in steps five and six, the model successfully merged the split and looping steps of the previous experiment back into a meaningful incident summary following the reports from top to bottom with accurate level of detail. In conclusion, the results suggest a causal relationship in which increased input detail is associated with improved accuracy and greater detail in the generated output.

V. Experiment Execution

Based on previously conducted experiments, each benchmark evaluation is prompted with the highest level of context detail in combination with the maximum data processing capabilities for each model, shown in Table V, to achieve the best possible result for each model. The referenced sources for each brief incident description match the sources provided by MITRE as foundation for their attack flows.

A. SolarWinds

In September 2019, malicious actors gained initial access to the internal development process of SolarWinds

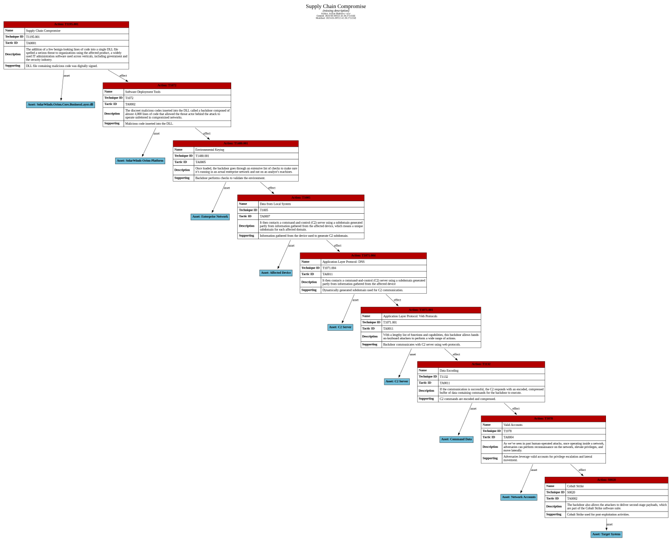
Supply Chain Compromise
(missing description)
Action: T1195.001
Name | Supply Chain Compromise
Technique ID | T1195.001
Tactic ID | TA0001
Description | The addition of a few benign-looking lines of code into a single DLL file spelled a serious threat to organizations using the affected product, a widely used IT administration software used across verticals, including government and the security industry.
Supporting | DLL file containing malicious code was digitally signed.
asset
effect
Asset: SolarWinds.Orion.Core.BusinessLayer.dll
Action: T1072
Name | Software Deployment Tools
Technique ID | T1072
Tactic ID | TA0002
Description | The discreet malicious codes inserted into the DLL called a backdoor composed of almost 4,000 lines of code that allowed the threat actor behind the attack to operate unfettered in compromised networks.
Supporting | Malicious code inserted into the DLL.
Asset: SolarWinds Orion Platform
Action: T1480.001
Name | Environmental Keying
Technique ID | T1480.001
Tactic ID | TA0005
Description | Once loaded, the backdoor goes through an extensive list of checks to make sure it's running in an actual enterprise network and not on an analyst's machines.
Supporting | Backdoor performs checks to validate the environment.
Asset: Enterprise Network
Action: T1005
Name | Data from Local System
Technique ID | T1005
Tactic ID | TA0007
Description | It then contacts a command-and-control (C2) server using a subdomain generated partly from information gathered from the affected device, which means a unique subdomain for each affected domain.
Supporting | Information gathered from the device used to generate C2 subdomain.
Asset: Affected Device
Action: T1071.004
Name | Application Layer Protocol: DNS
Technique ID | T1071.004
Tactic ID | TA0011
Description | It then contacts a command-and-control (C2) server using a subdomain generated partly from information gathered from the affected device
Supporting | Dynamically generated subdomain used for C2 communication.
Asset: C2 Server
Action: T1071.001
Name | Application Layer Protocol: Web Protocols
Technique ID | T1071.001
Tactic ID | TA0011
Description | With a lengthy list of functions and capabilities, this backdoor allows hands-on-keyboard attackers to perform a wide range of actions.
Supporting | Backdoor communicates with C2 server using web protocols.
Asset: C2 Server
Action: T1132
Name | Data Encoding
Technique ID | T1132
Tactic ID | TA0011
Description | If the communication is successful, the C2 responds with an encoded, compressed buffer of data containing commands for the backdoor to execute.
Supporting | C2 commands are encoded and compressed.
Asset: Command Data
Action: T1078
Name | Valid Accounts
Technique ID | T1078
Tactic ID | TA0004
Description | As we've seen in past human-operated attacks, once operating inside a network, adversaries can perform reconnaissance on the network, elevate privileges, and move laterally.
Supporting | Adversaries leverage valid accounts for privilege escalation and lateral movement.
Asset: Network Accounts
Action: S0020
Name | Cobalt Strike
Technique ID | S0020
Tactic ID | TA0002
Description | The backdoor also allows the attackers to deliver second-stage payloads, which are part of the Cobalt Strike software suite.
Supporting | Cobalt Strike used for post-exploitation activities.
Asset: Target System

Fig. 7. Attack flow diagram of SolarWinds incident with multiple MITRE references and multimodality

Orion[4]. The platform is a network management and monitoring tool and is distributed in US government agencies and S&P 500 companies [17]. Disguised as employees, an arbitrary code in the form of a backdoor mechanism was hidden inside the dynamic link library (DLL) "Solar-Winds.Orion.Core.BusinessLayer.dll", and equipped with an Orion official digital certificate, to bypass security scanners [18]. In March 2020, the malware component, later named SunBurst, was distributed within an automated update package to global customers.

Although the initial breach occurred at Orion, the actual targets of the attack were their customers. Once the update is deployed, the malware is installed as a windows service with the same elevated rights as the software, and is then calling the hidden lines of code, with the first objective to conduct reconnaissance and verify whether it is located on a real or a test environment and what security measurements such as antivirus or endpoint detection are in place. Secondly, a connection to the command and control server (C2) is established to direct and manage further attacks [19].

From this point onwards, the attackers were able to gradually increase their access and control of the system while still staying undetected. Outbound firewall rules were manipulated to bypass additional security layers, user credentials were obtained, connected systems were spied on, and with elevated rights, the backdoor mechanism was camouflaged to further prevent detection.

Now being deeply nested within the affected organisation and able to move across the whole system, the malicious actors installed additional DLLs on several machines on the system, each veiled as windows applications and with unique names to prevent easy detection of affected systems.

Ultimately, the attackers managed to disable all event logs and security hurdles during ongoing attacks and stole vast amounts of sensitive data.

a) Result: Tables X, XI, and XII show the averaged result for each model with the individual result shown in the Appendix. The Solarwinds incident produces a unique result with nearly identical numbers across four models, which are a direct result of the high-quality CTI report. Especially interesting for this case is the evaluation of the relationship metrics, as the DeepSeek model performs significantly better that its competitors, for the same number of true positive actions.

TABLE X
AVERAGE ACTION PREDICTION FOR SOLARWINDS INCIDENT

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	**0.316**	**0.632**	**0.387**	**0.48**
Gemini	0.294	0.615	0.361	0.455
LLaMa	**0.316**	**0.632**	**0.387**	**0.48**
GPT	**0.316**	**0.632**	**0.387**	**0.48**
Mistral	**0.316**	**0.632**	**0.387**	**0.48**

TABLE XI
AVERAGE ASSET PREDICTION FOR SOLARWINDS INCIDENT

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0.157	0.181	0.545	0.272
Gemini	0.162	0.186	**0.551**	0.279
LLaMa	0.039	0.053	0.122	0.073
GPT	**0.192**	**0.235**	0.514	**0.322**
Mistral	0.094	0.111	0.432	0.166

TABLE XII
AVERAGE RELATIONSHIP PREDICTION FOR SOLARWINDS INCIDENT

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	**0.265**	0.47	**0.378**	**0.419**
Gemini	0.173	0.399	0.233	0.294
LLaMa	0.122	0.388	0.151	0.217
GPT	0.21	**0.53**	0.258	0.347
Mistral	0.165	0.381	0.225	0.283

B. Sony Malware

The Sony Pictures malware attack of 2014 is a prime example of a hacking attack with the sole purpose of inflicting as much damage as possible without particularly sophisticated means. As a victim of a phishing attack, a Sony employee carelessly clicked a malicious link and installed malware that created a network file system (NFS) allowing external file access over network. The programme then elevates the NFS rights to unlimited access for the network, enabling the attacker to get full remote access of the machine.

Following the hijacking of the entry point, the malware is spreading laterally throughout the entire organisation, by abusing the internal network, where additional malware is installed on each affected machine [20].

a) Result: Tables XIII, XIV, and XV show the averaged result for each model with the individual result shown in the Appendix. Each evaluation object produces different top

[4] https://www.solarwinds.com/orion-platform; Accessed: 23.03.2025

performers, whereas the Mistral model excels in the asset prediction but completely fails in the action prediction, resulting in no relevant graph and therefore no correctly predicted relationships.

TABLE XIII
AVERAGE ACTION PREDICTION FOR SONY MALWARE INCIDENT

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	**0.176**	**0.321**	**0.277**	**0.296**
Gemini	0.067	0.14	0.115	0.126
LLaMa	0.122	0.304	0.169	0.217
GPT	0.126	0.249	0.2	0.22
Mistral	0	0	0	0

TABLE XIV
AVERAGE ASSET PREDICTION FOR SONY MALWARE INCIDENT

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0.038	0.07	0.073	0.071
Gemini	0.083	0.137	0.17	0.151
LLaMa	0.126	0.244	0.206	0.223
GPT	0.054	0.108	0.093	0.098
Mistral	**0.21**	**0.357**	**0.321**	**0.338**

TABLE XV
AVERAGE RELATIONSHIP PREDICTION FOR SONY MALWARE INCIDENT

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0.084	0.18	**0.135**	0.154
Gemini	0	0	0	0
LLaMa	**0.099**	**0.271**	0.134	**0.18**
GPT	0.061	0.138	0.099	0.113
Mistral	0	0	0	0

C. Target POS Breach

In a directed attack against Target, the attackers first performed a basic reconnaissance of the target's network and system configuration through simple means such as search engines. The research revealed a small maintenance supply vendor with access to the internal network and non-existing cyber security measurements [21]. Through phishing attacks and vendor infiltration, hackers obtained Active Directory (AD) credentials for the target network and were now able to move across the network. The attacker is reported to have no intention of targeting the point-of-sales (POS) system, but even though the IDS system targets reported unusual behaviour, no action was taken and the attacker was given time to develop, test and install malware on a large number of POS systems. The malware checked in a seven-hour cycle whether or not the stores are open (between 10 AM and 5 PM), and started to dump credit card details of customers on an internal server, where it was then collected by the hacker.

a) Result: Tables XVI, XVII, and XVIII show the averaged result for each model with the individual result shown in the Appendix. The action prediction is equally devastating across all models, but once again proves the Mistral model, with its extended input token limit, the best performance in asset prediction, and thus also relationship prediction.

TABLE XVI
AVERAGE ACTION PREDICTION FOR TARGET POS INCIDENT

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0.047	0.123	0.071	0.09
Gemini	0.049	0.132	0.071	0.093
LLaMa	**0.095**	**0.222**	**0.143**	**0.174**
GPT	0.082	0.219	0.114	0.15
Mistral	0.049	0.142	0.071	0.094

TABLE XVII
AVERAGE ASSET PREDICTION FOR TARGET POS INCIDENT

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0	0	0	0
Gemini	0	0	0	0
LLaMa	0	0	0	0
GPT	0.017	0.025	0.04	0.031
Mistral	**0.158**	**0.2**	**0.384**	**0.26**

TABLE XVIII
AVERAGE RELATIONSHIP PREDICTION FOR TARGET POS INCIDENT

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0	0	0	0
Gemini	0	0	0	0
LLaMa	0	0	0	0
GPT	0.007	0.018	0.011	0.013
Mistral	**0.036**	**0.094**	**0.057**	**0.07**

D. Tesla Kubernetes Breach

Due to insufficient security measurements and no password protection for the Tesla Kubernetes console, hackers were able to easily read the Tesla Amazon Web Service (AWS) credentials in plain text. In addition to the exposure of the data saved on the AWS system, the hackers used their access to create a new container in Kubernetes, using the computing power provided by AWS. The container was then used to run crypto-mining software and was integrated into a private mining pool.

a) Result: Tables XIX, XX, and XXI show the averaged result for each model with the individual result shown in the Appendix. Although the Gemini model correctly predicts more than a quarter of the relevant actions, it makes a single correct prediction between each. The LLaMa implementation, even if no correctly predicted asset, produces the highest score for the graph generation metric, which means the best resulting graph.

TABLE XIX
AVERAGE ACTION PREDICTION FOR TESLA KUBERNETES BREACH INCIDENT

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0.193	0.336	0.311	0.322
Gemini	**0.273**	**0.43**	**0.422**	**0.425**
LLaMa	0.162	0.38	0.222	0.279
GPT	0.135	0.25	0.222	0.234
Mistral	0.036	0.071	0.067	0.068

TABLE XX
Average asset prediction for Tesla Kubernetes Breach incident

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0.022	0.025	0.1	0.04
Gemini	**0.033**	**0.034**	**0.3**	**0.061**
LLaMa	0	0	0	0
GPT	0.025	0.029	0.1	0.044
Mistral	0	0	0	0

TABLE XXI
Average relationship prediction for Tesla Kubernetes Breach incident

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0.106	0.184	**0.198**	0.19
Gemini	0.057	0.086	0.143	0.107
LLaMa	**0.122**	**0.274**	0.182	**0.218**
GPT	0.078	0.143	0.15	0.145
Mistral	0.024	0.04	0.055	0.046

E. ToolShell

ToolShell represents an advanced SharePoint exploit combining ulnerabilities—CVE-2025-49706 (Authentication Bypass) and CVE-2025-49704 (Arbitrary File Write) to bypass authentication mechanisms and remotly execute arbitrary code on vulnerable and outdated servers. With the malicious shell execution, the attacker was able to extract secret cryptographic keys from the configuration files that are used for persistent access and total disclosure of affected SharePoint servers [22].

a) Result: Tables XXII, XXIII, and XXIV show the averaged result for each model with the individual result shown in the Appendix. Once more, the Gemini model performs best in the action prediction, but proves lacking in the graph-building relationship component. The best result for this incident is achieved by the implementation of GPT with constant results across all metrics.

TABLE XXII
Average action prediction for ToolShell incident

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0.155	0.293	0.244	0.266
Gemini	**0.235**	**0.476**	**0.333**	**0.377**
LLaMa	0.088	0.202	0.133	0.16
GPT	0.215	0.416	0.311	0.353
Mistral	0.088	0.317	0.111	0.162

TABLE XXIII
Average asset prediction for ToolShell incident

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0.1	0.133	0.133	0.133
Gemini	0	0	0	0
LLaMa	0	0	0	0
GPT	**0.133**	**0.15**	**0.171**	**0.16**
Mistral	0	0	0	0

TABLE XXIV
Average relationship prediction for ToolShell incident

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0.102	0.221	0.156	0.182
Gemini	0.014	0.033	0.022	0.027
LLaMa	0.043	0.12	0.063	0.083
GPT	**0.144**	0.284	**0.226**	**0.248**
Mistral	0.074	**0.295**	0.091	0.137

F. Turla-Carbon

Carbon is a sophisticated backdoor and framework developed and used by Turla, a cyber espionage threat group attributed to Russia's Federal Security Service (FSB). Carbon was selectively deployed to target organisations related to government and foreign affairs, with a particular focus on Central Asia [23], and once deployed, settles inside the systems by creating Windows services, naming them according to the existing naming scheme to operate undetected. Following the initial infection, the malware is carrying out excessive network reconnaissance to move lateral through the system and unnoticeably extract valuable data through C2 communication [23].

a) Result: Tables XXV, XXVI, and XXVII show the averaged result for each model with the individual result shown in the Appendix. Compared to many close results in previous incidents, Turla-Carbon shows a clear top performer in the DeepSeek model with extraordinary results for asset prediction.

TABLE XXV
Average action prediction for Turla-Carbon incident

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0.19	0.579	0.22	0.319
Gemini	**0.198**	0.601	**0.228**	**0.331**
LLaMa	0.092	0.381	0.108	0.168
GPT	0.189	**0.606**	0.216	0.319
Mistral	0.1	0.375	0.12	0.182

TABLE XXVI
Average asset prediction for Turla-Carbon incident

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	**0.476**	**0.786**	**0.514**	**0.621**
Gemini	0.274	0.281	0.408	0.315
LLaMa	0.077	0.2	0.092	0.125
GPT	0.211	0.399	0.218	0.28
Mistral	0.027	0.047	0.267	0.051

TABLE XXVII
Average relationship prediction for Turla-Carbon incident

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	**0.191**	**0.524**	**0.229**	**0.318**
Gemini	0.096	0.25	0.114	0.157
LLaMa	0.012	0.067	0.014	0.023
GPT	0.095	0.322	0.11	0.162
Mistral	0.037	0.143	0.046	0.071

G. Uber

As a result of supply chain access, where an external Uber contractor had their account compromised, adverseries gained access to Uber's internal VPN infrastructure [24]. Following the initial connection, the attackers achieved admin access due to discovery of hardcoded credentials in internal network files, which allowed for lateral movement across AWS, Google Cloud Platform, Google Drive, Slack workspace and others. Although this time no user data was affected, the incident describes already the third breach following the implementation of hardcoded credentials that resulted in leakage of internal financial information [25].

a) Result: Tables XXVIII, XXIX, and XXX show the averaged result for each model with the individual result shown in the Appendix. Once more the Gemini 2.5 flash model performs the best on the Uber datasets with half the actions and half the relations correctly defined, the output equals a good reconstruction of the attack kill chain.

TABLE XXVIII
AVERAGE ACTION PREDICTION FOR UBER INCIDENT

Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
DeepSeek	0.28	0.405	0.467	0.432
Gemini	**0.454**	**0.542**	**0.733**	**0.616**
LLaMa	0.091	0.167	0.167	0.167
GPT	0.253	0.381	0.4	0.39
Mistral	0.032	0.054	0.067	0.059

TABLE XXIX
AVERAGE ASSET PREDICTION FOR UBER INCIDENT

Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
DeepSeek	0.463	0.477	**0.913**	0.624
Gemini	0.223	0.244	0.711	0.361
LLaMa	**0.596**	0.671	0.809	**0.731**
GPT	0.545	**0.68**	0.734	0.702
Mistral	0.222	0.3	0.408	0.344

TABLE XXX
AVERAGE RELATIONSHIP PREDICTION FOR UBER INCIDENT

Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
DeepSeek	0.254	0.31	0.541	0.394
Gemini	**0.357**	**0.399**	**0.785**	**0.52**
LLaMa	0.132	0.19	0.292	0.23
GPT	0.286	0.357	0.53	0.423
Mistral	0.022	0.031	0.067	0.042

H. WhisperGate

WhisperGate was a destructive malware campaign deployed on Ukrainian systems at the beginning of the Russio-Ukrainian war in 2022. Although the malware follows the design patterns and mechanisms of traditional ransomware, it lacked the ability to recover any of the affected data, ruling out any form of financially motivated blackmail [26]. The attack was preceded by the initial acquisition of credentials that led to continued network access and lateral movement across multiple systems [27], ultimately causing significant damage to the Ukrainian infrastructure.

a) Result: Tables XXXI, XXXII, and XXXIII show the averaged result for each model with the individual result shown in the Appendix. The overall performance of all models is devastating, with rarely a correct prediction in any model. Especially the asset evaluation found no match in any model prediction.

TABLE XXXI
AVERAGE ACTION PREDICTION FOR WHISPERGATE INCIDENT

Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
DeepSeek	0.043	0.136	**0.059**	0.082
Gemini	0.039	0.104	**0.059**	0.075
LLaMa	**0.045**	**0.167**	**0.059**	**0.087**
GPT	**0.045**	0.162	**0.059**	0.086
Mistral	0.034	0.108	0.047	0.065

TABLE XXXII
AVERAGE ASSET PREDICTION FOR WHISPERGATE INCIDENT

Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
DeepSeek	0	0	0	0
Gemini	0	0	0	0
LLaMa	0	0	0	0
GPT	0	0	0	0
Mistral	0	0	0	0

TABLE XXXIII
AVERAGE RELATIONSHIP PREDICTION FOR WHISPERGATE INCIDENT

Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
DeepSeek	0.025	0.065	**0.04**	0.05
Gemini	0.015	0.031	0.03	0.03
LLaMa	**0.029**	**0.1**	**0.04**	**0.057**
GPT	0.028	0.083	**0.04**	0.054
Mistral	0.021	0.063	0.031	0.041

I. Swift

In a low-budget attack in 2016, hackers took advantage of Bangladesh lacking firewall system to gain access to elevated SWIFT credentials. With the gained access and distraction from previously dispatched malware affecting the daily business, they executed multiple fraudulent SWIFT payment orders. Due to weekend closing hours and global time zones, the attacker was able to successfully extract $81 million.

a) Result: Tables XXXIV, XXXV, and XXXVI show the averaged result for each model with the individual result shown in the Appendix. For the final incident, DeepSeek is able to show the best results for the action as well as the relationship evaluation. Although the results for Mistral are not far behind, Mistral was only able to produce three results in over 20 attempts and is the first model to fail the task without any result at all.

VI. RESULT ANALYSIS

The following section conducts a detailed summary and analysis of the experiment results for the DeepSeek, Gemini, LLaMa, GPT, and Mistral models. In general were the LLaMa, GPT and DeepSeek APIs very reliable and easy to use,

TABLE XXXIV
AVERAGE ACTION PREDICTION FOR SWIFT INCIDENT

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	**0.283**	**0.415**	**0.467**	**0.439**
Gemini	0.165	0.293	0.267	0.279
LLaMa	0.091	0.167	0.167	0.167
GPT	0.103	0.171	0.200	0.183
Mistral	0.148	0.3	0.222	0.255

TABLE XXXV
AVERAGE ASSET PREDICTION FOR SWIFT INCIDENT

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	0	0	0	0
Gemini	0	0	0	0
LLaMa	0	0	0	0
GPT	**0.075**	**0.086**	**0.150**	**0.109**
Mistral	0	0	0	0

while the Gemini model was drastically capped by hourly and minutely request limits. The Mistral thinking model often took multiple minutes to respond and was the only one to not produce any result in rare cases.

A. MITRE Mapping

Figure 8 shows the average resulting metric for the nine incidents previously described. The result of mapping the CTI content to the single MITRE techniques of the five models is consistent across all four metrics, where Mistral performs the worst with an average of 9% correctly matched actions and only 21% true matches of all predictions made. That result is doubled by the Gemini model with accuracy 20% as the top performer in the action evaluation, closely followed by the competitor DeepSeek- and GPT models. While the LLaMa implementation stagnates in the midfield, the Goolges Gemini model achieves the highest amount of predicted actions, along with the best measured information depth.

The best result across all models is achieved for the SolarWinds incident, while the WhisperGate CTI reports provide the worst result, as Fig. 9 shows for the incident average prediction values. A content analysis of the respective reveals an already partly conducted mapping to MITRE techniques for the Solarwinds incident [19], which can also be observed in Table X with near-equal action predictions for each model. Looking at the WhisperGate content offers an explanation for the low-input capacity models for GPT, LLaMa, and DeepSeek, as they were only able to process general incident descriptions with low technical depth [27], while Gemini and Mistral had access to more precise content [26]. Compared to the Uber incident, the second highest rated incident, the WhisperGate CTI contained large, unstructured, text with little detailed headers and subheaders.

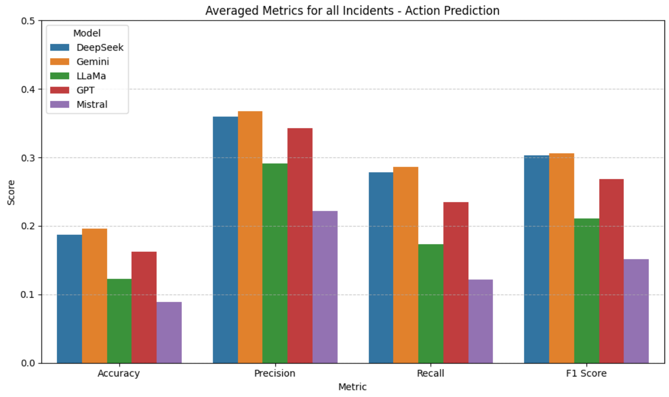

Fig. 8. Averaged Metrics for all Incidents - Action Prediction

TABLE XXXVI
AVERAGE RELATIONSHIP PREDICTION FOR SWIFT INCIDENT

Model	Accuracy	Precision	Recall	F1 Score
DeepSeek	**0.167**	**0.214**	**0.427**	**0.285**
Gemini	0.097	0.139	0.24	0.176
LLaMa	0.067	0.1	0.167	0.125
GPT	0.082	0.112	0.224	0.149
Mistral	0.121	0.207	0.222	0.214

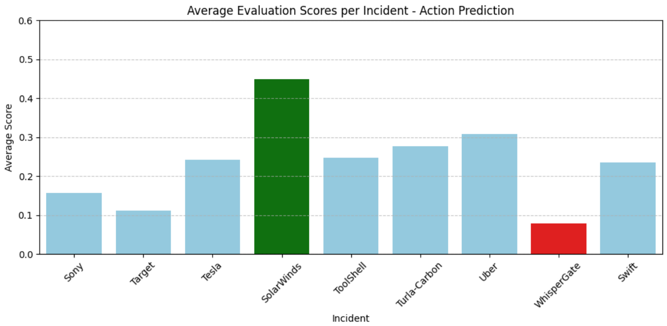

Fig. 9. Average Evaluation Scores per Incident - Action Prediction

B. Context Analysis

In accordance with the result of the previous action prediction, show the graph in Fig. 10 a nearly same landscape. A most noticeable difference is visible in the performance of the Gemini model, previously top performer, achieving similar results as Mistral and LLaMa. Not only with a 9% prediction accuracy, but also with a drastic overpredictiong of non-existant assets, failed the Google model in the context understanding rubic. DeepSeek and GPT on the other hand consistently perform and the same relative, but nevertheless reduced levels. This decrease in general performance proves an increased difficulty in predicting supportive objects compared to the general assignment of MITRE techniques.

Once more, the WhisperGate CTI fails to provide an understandable context for each model, with not a single correctly assessed asset (Fig. 11). On the other hand excels the Uber incident with a drastic outlier in correctly predicted assets. A view at the used reports reveals once more a very structured layout that gets preserved when converted to Markdown language and passed to the model. The writing style of the mainly used author is concisely describing different phases of the attack and how each phase was enabled [24], which leads

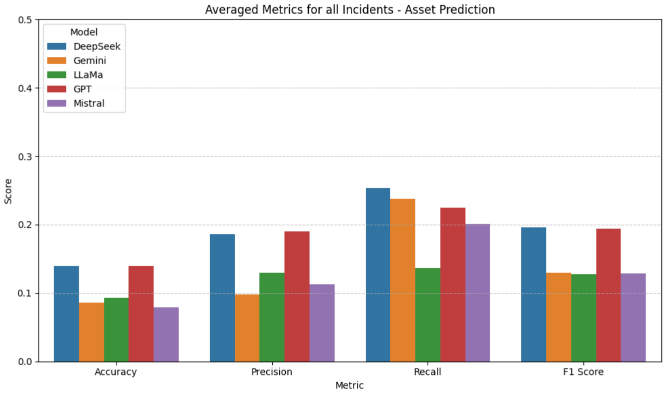

Fig. 10. Averaged Metrics for all Incidents - Asset Prediction

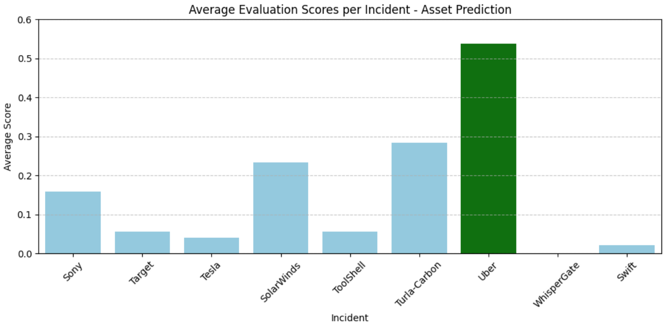

Fig. 11. Average Evaluation Scores per Incident - Asset Prediction

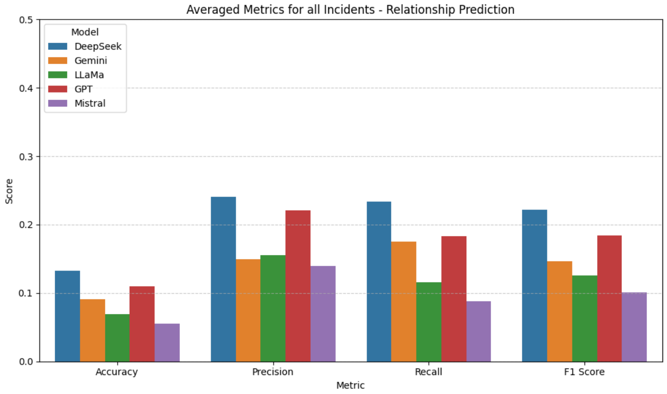

Fig. 12. Averaged Metrics for all Incidents - Relationship Prediction

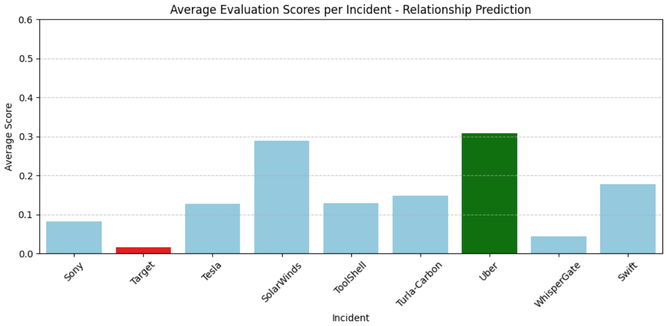

Fig. 13. Average Evaluation Scores per Incident - Relationship Prediction

to the high performance of all models, not only for the asset and context evaluation, but also for the technique mapping.

C. Graph Generation

The relationship evaluation in Fig. 12 compares the ability of each model not only to predict single components, as in the action and asset evaluation, but also to correctly build attack graphs and identify the order of actions and action-asset relations. Once more, the versions of DeepSeek and GPT compete for the rank of top-performer while LLaMa and Mistral provide not competitive result. With an accuracy of 13%, DeepSeek is able to recreate approximately one-tenth of the expert graph implementations, while the F1 score of 22% proves only a slightly better result for missed and overpredicted components. Directly corresponding to the best and worst interpretated CTI reports from the previous sections, SolarWinds and Uber offer the best graph reconstructions, while Target, WhisperGate and Sony, with less than 10% reconstructed content, fail miserably. With the evaluation system that requires correctly predicted actions and assets in order to achieve correct relationship predictions, this outcome is exactly as expected and proves the correctness of previous evaluations.

D. Benchmark

For a final conclusion on the current top performing large language model to automatically compute threat intelligence, all previous metric averages are combined and once again averaged. The result in Fig. 14 shows that DeepSeek-v3 is the clear top performer with 24% in the F1 score and on average 15% accurately predicted objects, closely followed by GPT-4o and Gemini 2.5 Flash. Although Gemini performed exceedingly well in the mapping of CTI content and MITRE ids, it failed in the context analyses and graph generation, resulting in only moderate overall results. The result is underlined by Fig. 15 and the average standard deviation per model prediction, calculated based on all experiment results

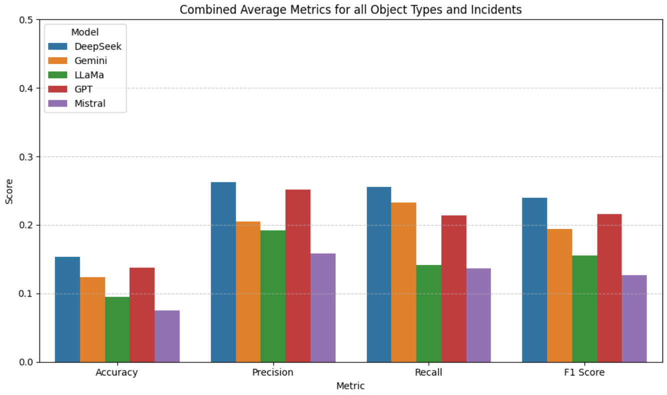

Fig. 14. Combined Average Metrics for all Object Types and Incidents

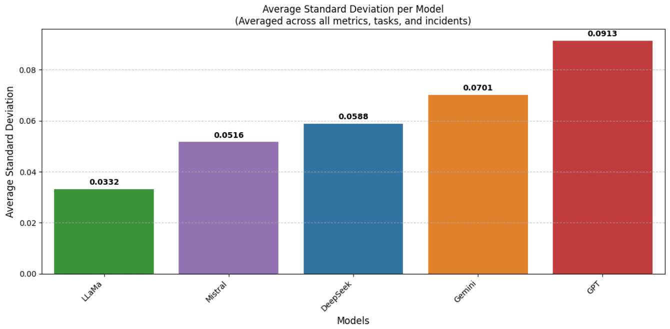

Fig. 15. Average Standard Deviation per Model (Averaged across all metrics, tasks, and incidents)

listed in the Apendix. DeepSeek increases its lead with low variance in prediction, while especially the GPT model shows an average deviation of nearly 10% for the same CTI reports in the same evaluation rubric and thus a very high fluctuation. This concludes DeepSeek-v3 as the top performer in this benchmark comparison for automated cyber threat intelligence processing, followed by a shared second place of Google Gemini 2.5 Flash and OpenAI GPT-4o.

VII. Conclusion

With an accuracy of only 15% of the best model to correctly interpret the CTI reports, none of the models can be rated to solve the task to automatically reconstruct CTI reports sufficiently with value for cybersecurity professionals. However, it shows great promise for the DeepSeek model to provide usable results in the future. Especially considering the major API restrictions that limit the input tokens per request to 4096 (compare Table V), as proven in the Target incident prediction in Tables XVII and XVIII resulting in the best results for the otherwise lowest ranked Mistral model with extended input window.

Furthermore, creates the implemented code foundation and evaluation framework a realistic and easily adaptive setup for future evaluation of improved API access and models.

References

[1] AtOnce, "41 microsoft 365 statistics and facts in 2025," https://atonce.com/learn/microsoft-365-statistics?utm_source=chatgpt.com, 2025, accessed: 2025-7-11.

[2] J. Stratton, *An Introduction to Microsoft Copilot*. Berkeley, CA: Apress, 2024, pp. 19–35. [Online]. Available: https://doi.org/10.1007/979-8-8688-0447-2_2

[3] MITRE, "Attack flow v3.0.0," https://center-for-threat-informed-defense.github.io/attack-flow/, 2025, accessed: 2025-7-11.

[4] H. Cuong Nguyen, S. Tariq, M. Baruwal Chhetri, and B. Quoc Vo, "Towards effective identification of attack techniques in cyber threat intelligence reports using large language models," in *Companion Proceedings of the ACM on Web Conference 2025*, ser. WWW '25. ACM, May 2025, p. 942–946. [Online]. Available: http://dx.doi.org/10.1145/3701716.3715469

[5] A. Virkud, M. A. Inam, A. Riddle, J. Liu, G. Wang, and A. Bates, "How does endpoint detection use the {mitre}{att&ck} framework?" in *33rd USENIX Security Symposium (USENIX Security 24)*, 2024, pp. 3891–3908.

[6] M. Zambianco, C. Facchinetti, and D. Siracusa, "A proactive decoy selection scheme for cyber deception using mitre att&ck," *Computers & Security*, vol. 148, p. 104144, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404824004498

[7] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong *et al.*, "A survey of large language models," *arXiv preprint arXiv:2303.18223*, vol. 1, no. 2, 2023.

[8] M. Danilevsky, K. Qian, R. Aharonov, Y. Katsis, B. Kawas, and P. Sen, "A survey of the state of explainable ai for natural language processing. arxiv," *arXiv preprint arXiv:2010.00711*, 2020.

[9] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[10] A. Chandra, L. Tünnermann, T. Löfstedt, and R. Gratz, "Transformer-based deep learning for predicting protein properties in the life sciences," *eLife*, vol. 12, p. e82819, jan 2023. [Online]. Available: https://doi.org/10.7554/eLife.82819

[11] D. Ristea, V. Mavroudis, and C. Hicks, "Ai cyber risk benchmark: Automated exploitation capabilities," *arXiv preprint arXiv:2410.21939*, 2024.

[12] Y. Zhu, A. Kellermann, D. Bowman, P. Li, A. Gupta, A. Danda, R. Fang, C. Jensen, E. Ihli, J. Benn *et al.*, "Cve-bench: A benchmark for ai agents' ability to exploit real-world web application vulnerabilities," *arXiv preprint arXiv:2503.17332*, 2025.

[13] M. Kouremetis, M. Dotter, A. Byrne, D. Martin, E. Michalak, G. Russo, M. Threet, and G. Zarrella, "Occult: Evaluating large language models for offensive cyber operation capabilities," *arXiv preprint arXiv:2502.15797*, 2025.

[14] A. Dawson, R. Mulla, N. Landers, and S. Caldwell, "Airtbench: Measuring autonomous ai red teaming capabilities in language models," *arXiv preprint arXiv:2506.14682*, 2025.

[15] Microsoft, "Cyberbattlesim," https://github.com/microsoft/CyberBattleSim, 2025.

[16] G. Marvin, N. Hellen Raudha, D. Jjingo, and J. Nakatumba-Nabende, *Prompt Engineering in Large Language Models*. Springer, 01 2024, pp. 387–402.

[17] SolarWinds, "Orion modules are now self-hosted on the solarwinds platform," https://www.solarwinds.com/orion-platform, 2025, accessed: 2025-03-25.

[18] Microsoft, "Deep dive into the solorigate second-stage activation: From sunburst to teardrop and raindrop," https://www.microsoft.com/en-us/security/blog/2021/01/20/deep-dive-into-the-solorigate-second-stage-activation-from-sunburst-to-teardrop-and-raindrop/, 2021, accessed: 2025-03-25.

[19] ——, "Analyzing solorigate, the compromised dll file that started a sophisticated cyberattack, and how microsoft defender helps protect customers," https://www.microsoft.com/en-us/security/blog/2020/12/18/analyzing-solorigate-the-compromised-dll-file-that-started-a-sophisticated-cyberattack-and-how-microsoft-defender-helps-protect/, 2020, accessed: 2025-03-25.

[20] S. Gallagher, "Inside the "wiper" malware that brought sony pictures to its knees [update]," https://arstechnica.com/information-technology/2014/12/inside-the-wiper-malware-that-brought-sony-pictures-to-its-knees/, 2014, accessed: 2025-03-25.

[21] COMMITTEE ON COMMERCE, SCIENCE, AND TRANSPORTATION, "A "kill chain" analysis of the 2013 target data breach," *Unknown Journal*, 2014.

[22] Varonis Threat Labs, "Toolshell: A sharepoint rce chain actively exploited," https://www.varonis.com/blog/toolshell-sharepoint-rce, 2025, accessed: 2025-07-25.

[23] MITRE, "Carbon," https://attack.mitre.org/software/S0335/, 2025, accessed: 2025-07-25.

[24] CyberArk Blog Team, "Unpacking the uber breach," https://www.cyberark.com/resources/blog/unpacking-the-uber-breach, 2022, accessed: 2025-07-25.

[25] Uber, "Security update," https://www.uber.com/newsroom/security-update/, 2022, accessed: 2025-07-25.

[26] M. Microsoft Digital Security Unit (DSU) and Microsoft Threat Intelligence, "Destructive malware targeting ukrainian organizations," https://www.microsoft.com/en-us/security/blog/2022/01/15/destructive-malware-targeting-ukrainian-organizations/, 2022, accessed: 2025-07-25.

[27] N. Biasini, M. Chen, A. Karkins, A. Khodjibaev, C. Neal, and M. Olney, “Ukraine campaign delivers defacement and wipers, in continued escalation,” https://blog.talosintelligence.com/ukraine-campaign-delivers-defacement/, 2022, accessed: 2025-07-25.

APPENDIX A
FULL EXPERIMENT RESULTS

TABLE XXXVII
ACTION PREDICTION FOR SOLARWINDS INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
Gemini	11	7	20	0.289	0.611	0.355	0.449
	11	7	20	0.289	0.611	0.355	0.449
	11	7	20	0.289	0.611	0.355	0.449
	12	7	19	0.316	0.632	0.387	0.48
	11	7	20	0.289	0.611	0.355	0.449
LLaMa	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
GPT	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
Mistral	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48
	12	7	19	0.316	0.632	0.387	0.48

TABLE XXXVIII
ASSET PREDICTION FOR SOLARWINDS INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	6	29	5	0.15	0.171	0.545	0.261
	6	27	5	0.158	0.182	0.545	0.273
	6	27	5	0.158	0.182	0.545	0.273
	6	27	5	0.158	0.182	0.545	0.273
	6	26	5	0.162	0.188	0.545	0.279
Gemini	6	25	5	0.167	0.194	0.545	0.286
	7	26	5	0.184	0.212	0.583	0.311
	6	28	5	0.154	0.176	0.545	0.267
	7	29	5	0.171	0.194	0.583	0.292
	5	27	5	0.135	0.156	0.5	0.238
LLaMa	0	19	9	0	0	0	0
	2	17	6	0.08	0.105	0.25	0.148
	0	18	9	0	0	0	0
	1	18	8	0.037	0.053	0.111	0.071
	2	17	6	0.08	0.105	0.25	0.148
GPT	5	15	4	0.208	0.25	0.556	0.345
	5	15	5	0.2	0.25	0.5	0.333
	5	15	5	0.2	0.25	0.5	0.333
	5	21	5	0.161	0.192	0.5	0.278
Mistral	3	17	5	0.12	0.15	0.375	0.214
	2	19	7	0.095	0.095	1	0.174
	0	19	9	0	0	0	0
	4	15	4	0.174	0.211	0.5	0.296
	2	18	5	0.08	0.1	0.286	0.148

TABLE XXXIX
RELATIONSHIP PREDICTION FOR SOLARWINDS INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	17	21	28	0.258	0.447	0.378	0.41
	17	19	28	0.266	0.472	0.378	0.42
	17	19	28	0.266	0.472	0.378	0.42
	17	19	28	0.266	0.472	0.378	0.42
	17	18	28	0.27	0.486	0.378	0.425
Gemini	8	12	29	0.163	0.4	0.216	0.281
	9	13	29	0.176	0.409	0.237	0.3
	8	13	29	0.16	0.381	0.216	0.276
	11	15	28	0.204	0.423	0.282	0.338
	8	13	29	0.16	0.381	0.216	0.276
LLaMa	4	8	28	0.1	0.333	0.125	0.182
	6	8	28	0.143	0.429	0.176	0.25
	4	8	28	0.1	0.333	0.125	0.182
	5	8	28	0.122	0.385	0.152	0.217
	6	7	28	0.146	0.462	0.176	0.255
GPT	10	8	28	0.217	0.556	0.263	0.357
	10	8	28	0.217	0.556	0.263	0.357
	10	8	28	0.217	0.556	0.263	0.357
	9	11	28	0.188	0.45	0.243	0.316
Mistral	9	13	28	0.18	0.409	0.243	0.305
	8	14	28	0.16	0.364	0.222	0.276
	6	13	28	0.128	0.316	0.176	0.226
	10	13	28	0.196	0.435	0.263	0.328
	8	13	28	0.163	0.381	0.222	0.281

TABLE XL
ACTION PREDICTION FOR TESLA KUBERNETES BREACH INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	3	6	6	0.2	0.333	0.333	0.333
	3	6	6	0.2	0.333	0.333	0.333
	3	6	6	0.2	0.333	0.333	0.333
	2	6	7	0.133	0.25	0.222	0.235
	3	4	6	0.231	0.429	0.333	0.375
Gemini	5	5	4	0.357	0.5	0.556	0.526
	3	5	6	0.214	0.375	0.333	0.353
	4	4	5	0.308	0.5	0.444	0.471
	4	5	5	0.286	0.444	0.444	0.444
	3	6	6	0.2	0.333	0.333	0.333
LLaMa	2	4	7	0.154	0.333	0.222	0.267
	2	3	7	0.167	0.4	0.222	0.286
	2	4	7	0.154	0.333	0.222	0.267
	2	4	7	0.154	0.333	0.222	0.267
	2	2	7	0.182	0.5	0.222	0.308
GPT	1	7	8	0.062	0.125	0.111	0.118
	2	6	7	0.133	0.25	0.222	0.235
	2	7	7	0.125	0.222	0.222	0.222
	2	7	7	0.125	0.222	0.222	0.222
	3	4	6	0.231	0.429	0.333	0.375
Mistral	0	2	9	0	0	0	0
	1	8	8	0.059	0.111	0.111	0.111
	1	6	8	0.067	0.143	0.111	0.125
	1	9	8	0.056	0.1	0.111	0.105
	0	9	9	0	0	0	0

TABLE XLI
ASSET PREDICTION FOR TESLA KUBERNETES BREACH INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	0	10	2	0	0	0	0
	0	10	2	0	0	0	0
	0	11	2	0	0	0	0
	0	10	2	0	0	0	0
	1	7	1	0.111	0.125	0.5	0.2
Gemini	1	19	1	0.048	0.05	0.5	0.091
	1	15	1	0.059	0.062	0.5	0.111
	0	15	2	0	0	0	0
	0	14	2	0	0	0	0
	1	16	1	0.056	0.059	0.5	0.105
LLaMa	0	7	2	0	0	0	0
	0	7	2	0	0	0	0
	0	8	2	0	0	0	0
	0	7	2	0	0	0	0
	0	7	2	0	0	0	0
GPT	0	8	2	0	0	0	0
	0	8	2	0	0	0	0
	0	9	2	0	0	0	0
	0	9	2	0	0	0	0
	1	6	1	0.125	0.143	0.5	0.222
Mistral	0	5	2	0	0	0	0
	0	10	2	0	0	0	0
	0	10	2	0	0	0	0
	0	11	2	0	0	0	0
	0	10	2	0	0	0	0

TABLE XLII
Relationship prediction for Tesla Kubernetes Breach incident

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	3	11	9	0.13	0.214	0.25	0.231
	3	11	9	0.13	0.214	0.25	0.231
	3	12	10	0.12	0.2	0.231	0.214
	1	10	10	0.048	0.091	0.091	0.091
	2	8	10	0.1	0.2	0.167	0.182
Gemini	2	18	8	0.071	0.1	0.2	0.133
	2	17	10	0.069	0.105	0.167	0.129
	2	12	9	0.087	0.143	0.182	0.16
	0	7	9	0	0	0	0
	2	22	10	0.059	0.083	0.167	0.111
LLaMa	2	6	9	0.118	0.25	0.182	0.211
	2	4	9	0.133	0.333	0.182	0.235
	2	6	9	0.118	0.25	0.182	0.211
	2	6	9	0.118	0.25	0.182	0.211
	2	5	9	0.125	0.286	0.182	0.222
GPT	1	13	11	0.04	0.071	0.083	0.077
	2	11	10	0.087	0.154	0.167	0.16
	2	13	10	0.08	0.133	0.167	0.148
	2	13	10	0.08	0.133	0.167	0.148
	2	7	10	0.105	0.222	0.167	0.19
Mistral	0	3	12	0	0	0	0
	1	15	10	0.038	0.062	0.091	0.074
	1	11	10	0.045	0.083	0.091	0.0878
	1	17	10	0.036	0.056	0.091	0.069
	0	17	12	0	0	0	0

TABLE XLIII
Action prediction for Target POS incident

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	1	7	13	0.048	0.125	0.071	0.091
	1	8	13	0.045	0.111	0.071	0.087
	1	8	13	0.045	0.111	0.071	0.087
	1	8	13	0.045	0.111	0.071	0.087
	1	6	13	0.05	0.143	0.071	0.095
Gemini	1	7	13	0.048	0.125	0.071	0.091
	1	7	13	0.048	0.125	0.071	0.091
	1	6	13	0.05	0.143	0.071	0.095
	1	6	13	0.05	0.143	0.071	0.095
	1	7	13	0.048	0.125	0.071	0.091
LLaMa	2	7	12	0.095	0.222	0.143	0.174
	2	7	12	0.095	0.222	0.143	0.174
	2	7	12	0.095	0.222	0.143	0.174
	2	7	12	0.095	0.222	0.143	0.174
	2	7	12	0.095	0.222	0.143	0.174
GPT	1	8	13	0.045	0.111	0.071	0.087
	2	5	12	0.105	0.286	0.143	0.19
	2	5	12	0.105	0.286	0.143	0.19
	2	5	12	0.105	0.286	0.143	0.19
	1	7	13	0.048	0.125	0.071	0.091
Mistral	1	6	13	0.05	0.143	0.071	0.095
	1	8	13	0.045	0.111	0.071	0.087
	1	6	13	0.05	0.143	0.071	0.095
	1	4	13	0.056	0.2	0.071	0.105
	1	8	13	0.045	0.111	0.071	0.087

TABLE XLIV
Asset prediction for Target POS incident

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	0	9	5	0	0	0	0
	0	9	5	0	0	0	0
	0	9	5	0	0	0	0
	0	9	5	0	0	0	0
	0	9	5	0	0	0	0
Gemini	0	9	5	0	0	0	0
	0	9	5	0	0	0	0
	0	8	5	0	0	0	0
	0	8	5	0	0	0	0
	0	9	5	0	0	0	0
LLaMa	0	10	5	0	0	0	0
	0	10	5	0	0	0	0
	0	10	5	0	0	0	0
	0	10	5	0	0	0	0
	0	10	5	0	0	0	0
GPT	0	9	5	0	0	0	0
	0	7	5	0	0	0	0
	0	7	5	0	0	0	0
	0	7	5	0	0	0	0
	1	7	4	0.083	0.125	0.2	0.154
Mistral	4	6	3	0.308	0.4	0.571	0.471
	1	9	3	0.077	0.1	0.25	0.143
	1	9	3	0.077	0.1	0.25	0.143
	1	9	3	0.077	0.1	0.25	0.143
	3	7	2	0.25	0.3	0.6	0.4

TABLE XLV
RELATIONSHIP PREDICTION FOR TARGET POS INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	0	14	20	0	0	0	0
	0	15	20	0	0	0	0
	0	14	20	0	0	0	0
	0	15	20	0	0	0	0
	0	13	21	0	0	0	0
Gemini	0	14	20	0	0	0	0
	0	14	20	0	0	0	0
	0	12	21	0	0	0	0
	0	12	21	0	0	0	0
	0	14	20	0	0	0	0
LLaMa	0	10	18	0	0	0	0
	0	10	19	0	0	0	0
	0	10	19	0	0	0	0
	0	10	19	0	0	0	0
	0	10	19	0	0	0	0
GPT	0	9	20	0	0	0	0
	1	10	18	0.034	0.091	0.053	0.067
	0	5	19	0	0	0	0
	0	5	19	0	0	0	0
	0	12	20	0	0	0	0
Mistral	2	11	20	0.061	0.154	0.091	0.114
	1	15	20	0.028	0.062	0.048	0.054
	1	11	20	0.031	0.083	0.048	0.061
	1	8	20	0.034	0.111	0.048	0.067
	1	15	20	0.028	0.062	0.048	0.054

TABLE XLVI
ACTION PREDICTION FOR SONY MALWARE INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	4	7	9	0.2	0.364	0.308	0.333
	4	9	9	0.182	0.308	0.308	0.308
	4	6	9	0.211	0.4	0.308	0.348
	2	10	11	0.087	0.167	0.154	0.16
	4	7	9	0.2	0.364	0.308	0.333
Gemini	2	10	11	0.087	0.167	0.154	0.16
	1	7	12	0.05	0.125	0.077	0.095
	1	9	12	0.045	0.1	0.077	0.087
	2	10	11	0.087	0.167	0.154	0.16
LLaMa	2	5	11	0.111	0.286	0.154	0.2
	2	5	11	0.111	0.286	0.154	0.2
	2	5	11	0.111	0.286	0.154	0.2
	2	5	11	0.111	0.286	0.154	0.2
	3	5	10	0.167	0.375	0.231	0.286
GPT	3	9	10	0.136	0.25	0.231	0.24
	1	6	12	0.053	0.143	0.077	0.1
	3	9	10	0.136	0.25	0.231	0.24
	4	6	9	0.211	0.4	0.308	0.348
	2	8	11	0.095	0.2	0.154	0.174
Mistral	0	4	13	0	0	0	0
	0	2	13	0	0	0	0
	0	2	13	0	0	0	0
	0	5	13	0	0	0	0
	0	2	13	0	0	0	0

TABLE XLVII
ASSET PREDICTION FOR SONY MALWARE INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	1	11	9	0.048	0.083	0.1	0.091
	0	14	11	0	0	0	0
	0	11	15	0	0	0	0
	1	11	11	0.043	0.083	0.083	0.083
	2	9	9	0.1	0.182	0.182	0.182
Gemini	2	13	9	0.083	0.133	0.182	0.154
	1	8	9	0.056	0.111	0.1	0.105
	1	10	9	0.05	0.091	0.1	0.095
	3	11	7	0.143	0.214	0.3	0.25
LLaMa	2	7	9	0.111	0.222	0.182	0.2
	2	7	9	0.111	0.222	0.182	0.2
	2	7	9	0.111	0.222	0.182	0.2
	2	7	9	0.111	0.222	0.182	0.2
	3	6	7	0.188	0.333	0.3	0.316
GPT	0	13	11	0	0	0	0
	2	5	9	0.125	0.286	0.182	0.222
	2	11	9	0.091	0.154	0.182	0.167
	1	9	9	0.053	0.1	0.1	0.1
	0	10	11	0	0	0	0
Mistral	4	6	7	0.235	0.4	0.364	0.381
	4	4	5	0.308	0.5	0.444	0.471
	4	6	7	0.235	0.4	0.364	0.381
	4	6	8	0.222	0.4	0.333	0.364
	1	11	9	0.048	0.083	0.1	0.091

TABLE XLVIII
Relationship prediction for Sony Malware incident

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	4	15	20	0.103	0.211	0.167	0.186
	3	17	20	0.075	0.15	0.13	0.14
	2	11	20	0.061	0.154	0.091	0.114
	3	17	22	0.071	0.15	0.12	0.133
	4	13	20	0.108	0.235	0.167	0.195
Gemini	0	13	22	0	0	0	0
	0	8	23	0	0	0	0
	0	10	23	0	0	0	0
	0	12	22	0	0	0	0
LLaMa	3	9	22	0.088	0.25	0.12	0.162
	3	9	22	0.088	0.25	0.12	0.162
	3	9	22	0.088	0.25	0.12	0.162
	3	9	22	0.088	0.25	0.12	0.162
	5	9	21	0.143	0.357	0.192	0.25
GPT	3	17	17	0.081	0.15	0.15	0.15
	1	11	24	0.028	0.083	0.04	0.054
	4	17	20	0.098	0.19	0.167	0.178
	2	9	19	0.067	0.182	0.095	0.125
	1	11	23	0.029	0.083	0.042	0.056
Mistral	0	14	25	0	0	0	0
	0	5	25	0	0	0	0
	0	5	25	0	0	0	0
	0	12	25	0	0	0	0
	0	5	25	0	0	0	0

TABLE XLIX
Action prediction for ToolShell incident

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	3	4	6	0.231	0.429	0.333	0.375
	2	6	7	0.133	0.25	0.222	0.235
	2	5	7	0.143	0.286	0.222	0.25
	2	6	7	0.133	0.25	0.222	0.235
	2	6	7	0.133	0.25	0.222	0.235
Gemini	3	4	6	0.231	0.429	0.333	0.375
	3	1	6	0.3	0.75	0.333	0.462
	3	3	6	0.25	0.5	0.333	0.4
	3	3	6	0.25	0.5	0.333	0.4
	3	12	6	0.143	0.2	0.333	0.25
LLaMa	1	6	8	0.067	0.143	0.111	0.125
	1	5	8	0.071	0.167	0.111	0.133
	1	5	8	0.071	0.167	0.111	0.133
	1	4	8	0.077	0.2	0.111	0.143
	2	4	7	0.154	0.333	0.222	0.267
GPT	3	4	6	0.231	0.429	0.333	0.375
	3	5	6	0.214	0.375	0.333	0.353
	3	5	6	0.214	0.375	0.333	0.353
	3	3	6	0.25	0.5	0.333	0.4
	2	3	7	0.167	0.4	0.222	0.286
Mistral	1	1	8	0.1	0.5	0.111	0.182
	1	2	8	0.091	0.333	0.111	0.167
	1	3	8	0.083	0.25	0.111	0.154
	1	3	8	0.083	0.25	0.111	0.154
	1	3	8	0.083	0.25	0.111	0.154

TABLE L
Asset prediction for ToolShell incident

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	0	10	3	0	0	0	0
	0	10	3	0	0	0	0
	0	9	3	0	0	0	0
	6	3	3	0.5	0.667	0.667	0.667
	0	10	3	0	0	0	0
Gemini	0	14	3	0	0	0	0
	0	15	3	0	0	0	0
	0	14	3	0	0	0	0
	0	18	3	0	0	0	0
	0	28	3	0	0	0	0
LLaMa	0	9	3	0	0	0	0
	0	9	3	0	0	0	0
	0	8	3	0	0	0	0
	0	9	3	0	0	0	0
	0	9	3	0	0	0	0
GPT	0	9	3	0	0	0	0
	0	9	3	0	0	0	0
	6	2	1	0.667	0.75	0.857	0.8
	0	8	3	0	0	0	0
	0	9	3	0	0	0	0
Mistral	0	9	3	0	0	0	0
	0	8	3	0	0	0	0
	0	9	3	0	0	0	0
	0	8	3	0	0	0	0
	0	11	3	0	0	0	0

TABLE LI
RELATIONSHIP PREDICTION FOR TOOLSHELL INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	3	6	8	0.176	0.333	0.273	0.3
	2	11	17	0.067	0.154	0.105	0.125
	2	9	17	0.071	0.182	0.105	0.133
	4	11	17	0.125	0.267	0.19	0.222
	2	10	17	0.069	0.167	0.105	0.129
Gemini	0	9	8	0	0	0	0
	1	5	8	0.071	0.167	0.111	0.133
	0	7	8	0	0	0	0
	0	9	8	0	0	0	0
	0	20	8	0	0	0	0
LLaMa	1	11	18	0.033	0.083	0.053	0.065
	1	10	18	0.034	0.091	0.053	0.067
	1	10	18	0.034	0.091	0.053	0.067
	1	8	18	0.037	0.111	0.053	0.071
	2	7	17	0.077	0.222	0.105	0.143
GPT	2	9	8	0.105	0.182	0.2	0.19
	3	9	13	0.12	0.25	0.188	0.214
	7	9	13	0.241	0.438	0.35	0.389
	3	7	8	0.167	0.3	0.273	0.286
	2	6	15	0.087	0.25	0.118	0.16
Mistral	2	2	18	0.091	0.5	0.1	0.167
	2	4	18	0.083	0.333	0.1	0.154
	1	6	18	0.04	0.143	0.053	0.077
	2	6	18	0.077	0.25	0.1	0.143
	2	6	18	0.077	0.25	0.1	0.143

TABLE LII
ACTION PREDICTION FOR TURLA-CARBON INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	11	8	39	0.19	0.579	0.22	0.319
	11	8	39	0.19	0.579	0.22	0.319
	11	8	39	0.19	0.579	0.22	0.319
	11	8	39	0.19	0.579	0.22	0.319
	11	8	39	0.19	0.579	0.22	0.319
Gemini	11	7	39	0.193	0.611	0.22	0.324
	12	9	38	0.203	0.571	0.24	0.338
	12	8	38	0.207	0.6	0.24	0.343
	11	7	39	0.193	0.611	0.22	0.324
	11	7	39	0.193	0.611	0.22	0.324
LLaMa	5	9	45	0.085	0.357	0.1	0.156
	6	8	44	0.103	0.429	0.12	0.188
	5	10	45	0.083	0.333	0.1	0.154
	6	8	44	0.103	0.429	0.12	0.188
	5	9	45	0.085	0.357	0.1	0.156
GPT	11	7	39	0.193	0.611	0.22	0.324
	11	7	39	0.193	0.611	0.22	0.324
	10	7	40	0.175	0.588	0.2	0.299
	11	7	39	0.193	0.611	0.22	0.324
	11	7	39	0.193	0.611	0.22	0.324
Mistral	6	10	44	0.1	0.375	0.12	0.182
	6	10	44	0.1	0.375	0.12	0.182
	6	10	44	0.1	0.375	0.12	0.182
	6	10	44	0.1	0.375	0.12	0.182

TABLE LIII
ASSET PREDICTION FOR TURLA-CARBON INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	18	1	15	0.529	0.947	0.545	0.692
	6	14	22	0.143	0.3	0.214	0.25
	16	3	15	0.471	0.842	0.516	0.64
	16	3	11	0.533	0.842	0.593	0.696
	19	0	8	0.704	1	0.704	0.826
Gemini	25	10	21	0.446	0.714	0.543	0.617
	0	42	28	0	0	0	0
	42	0	0	1	1	1	1
	15	28	22	0.341	0.349	0.938	0.508
	0	31	28	0	0	0	0
LLaMa	1	18	21	0.025	0.053	0.045	0.049
	1	13	21	0.018	0.071	0.023	0.035
	11	4	21	0.306	0.733	0.344	0.468
	1	13	36	0.02	0.071	0.027	0.039
	1	13	21	0.018	0.071	0.023	0.035
GPT	0	18	27	0	0	0	0
	1	17	27	0.022	0.056	0.036	0.043
	16	1	11	0.571	0.941	0.593	0.727
	0	18	27	0	0	0	0
	18	0	21	0.462	1	0.462	0.632
Mistral	1	15	31	0.062	0.062	1	0.118
	0	17	32	0	0	0	0
	0	16	32	0	0	0	0
	2	14	28	0.045	0.125	0.067	0.087

TABLE LIV
RELATIONSHIP PREDICTION FOR TURLA-CARBON INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	19	15	57	0.209	0.559	0.25	0.345
	9	15	57	0.111	0.375	0.136	0.2
	20	15	57	0.217	0.571	0.26	0.357
	18	15	57	0.2	0.545	0.24	0.333
	20	15	57	0.217	0.571	0.26	0.357
Gemini	23	12	57	0.25	0.657	0.287	0.4
	0	16	55	0	0	0	0
	43	12	55	0.391	0.782	0.439	0.562
	14	19	57	0.156	0.424	0.197	0.269
	0	10	57	0	0	0	0
LLaMa	0	13	68	0	0	0	0
	0	8	66	0	0	0	0
	5	10	68	0.06	0.333	0.068	0.114
	0	8	66	0	0	0	0
	0	9	68	0	0	0	0
GPT	1	9	57	0.015	0.1	0.017	0.029
	3	19	57	0.038	0.136	0.05	0.073
	18	13	59	0.2	0.581	0.234	0.333
	1	8	57	0.015	0.111	0.017	0.03
	17	8	57	0.207	0.68	0.23	0.343
Mistral	3	20	66	0.034	0.13	0.043	0.065
	3	20	66	0.034	0.13	0.043	0.065
	3	19	66	0.034	0.136	0.043	0.066
	4	19	66	0.045	0.174	0.057	0.086

TABLE LV
ACTION PREDICTION FOR UBER INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	2	4	4	0.2	0.333	0.333	0.333
	3	4	3	0.3	0.429	0.5	0.462
	3	4	3	0.3	0.429	0.5	0.462
	4	4	2	0.4	0.5	0.667	0.571
	2	4	4	0.2	0.333	0.333	0.333
Gemini	5	2	1	0.625	0.714	0.833	0.769
	5	6	1	0.417	0.455	0.833	0.588
	5	5	1	0.455	0.5	0.833	0.625
	3	5	3	0.273	0.375	0.5	0.429
	4	2	2	0.5	0.667	0.667	0.667
LLaMa	1	5	5	0.091	0.167	0.167	0.167
	1	5	5	0.091	0.167	0.167	0.167
	1	5	5	0.091	0.167	0.167	0.167
	1	5	5	0.091	0.167	0.167	0.167
	1	5	5	0.091	0.167	0.167	0.167
GPT	4	2	2	0.5	0.667	0.667	0.667
	2	5	4	0.182	0.286	0.333	0.308
	2	4	4	0.2	0.333	0.333	0.333
	2	4	4	0.2	0.333	0.333	0.333
	2	5	4	0.182	0.286	0.333	0.308
Mistral	1	6	5	0.083	0.143	0.167	0.154
	0	5	6	0	0	0	0
	1	7	5	0.077	0.125	0.167	0.143
	0	8	6	0	0	0	0
	0	8	6	0	0	0	0

TABLE LVI
ASSET PREDICTION FOR UBER INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	5	5	2	0.5	0.5	1	0.667
	6	4	2	0.6	0.6	1	0.75
	5	7	1	0.417	0.417	1	0.588
	9	7	1	0.529	0.562	0.9	0.692
	4	9	2	0.267	0.308	0.667	0.421
Gemini	7	17	2	0.269	0.292	0.778	0.424
	6	18	2	0.231	0.25	0.75	0.375
	5	24	3	0.156	0.172	0.625	0.27
	11	25	2	0.306	0.306	1	0.468
	4	16	6	0.154	0.2	0.4	0.267
LLaMa	4	2	1	0.667	0.667	1	0.8
	5	1	1	0.833	0.833	1	0.909
	3	4	3	0.3	0.429	0.5	0.462
	5	2	1	0.625	0.714	0.833	0.769
	5	2	2	0.556	0.714	0.714	0.714
GPT	6	3	2	0.545	0.667	0.75	0.706
	5	3	1	0.556	0.625	0.833	0.714
	6	1	2	0.667	0.857	0.75	0.8
	5	3	3	0.455	0.625	0.625	0.625
	5	3	2	0.5	0.625	0.714	0.667
Mistral	2	6	3	0.182	0.25	0.4	0.308
	4	4	3	0.364	0.5	0.571	0.533
	4	4	2	0.4	0.5	0.667	0.571
	1	7	4	0.083	0.125	0.2	0.154
	1	7	4	0.083	0.125	0.2	0.154

TABLE LVII
RELATIONSHIP PREDICTION FOR UBER INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	2	8	4	0.143	0.2	0.333	0.25
	6	9	3	0.333	0.4	0.667	0.5
	4	9	3	0.25	0.308	0.571	0.4
	8	10	2	0.4	0.444	0.8	0.571
	2	8	4	0.143	0.2	0.333	0.25
Gemini	7	7	1	0.467	0.5	0.875	0.636
	6	12	1	0.316	0.333	0.857	0.48
	3	8	1	0.25	0.273	0.75	0.4
	8	16	3	0.296	0.333	0.727	0.457
	5	4	2	0.455	0.556	0.714	0.625
LLaMa	1	9	5	0.067	0.1	0.167	0.125
	3	9	5	0.176	0.25	0.375	0.3
	1	9	5	0.067	0.1	0.167	0.125
	3	9	5	0.176	0.25	0.375	0.3
	3	9	5	0.176	0.25	0.375	0.3
GPT	9	4	2	0.6	0.692	0.818	0.75
	2	10	4	0.125	0.167	0.333	0.222
	4	8	4	0.25	0.333	0.5	0.4
	4	9	4	0.235	0.308	0.5	0.381
	4	10	4	0.222	0.286	0.5	0.364
Mistral	1	11	5	0.059	0.083	0.167	0.111
	0	10	6	0	0	0	0
	1	13	5	0.053	0.071	0.167	0.1
	0	15	6	0	0	0	0
	0	15	6	0	0	0	0

TABLE LVIII
ACTION PREDICTION FOR WHISPERGATE INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	1	7	16	0.042	0.125	0.059	0.08
	1	6	16	0.043	0.143	0.059	0.083
	1	7	16	0.042	0.125	0.059	0.08
	1	6	16	0.043	0.143	0.059	0.083
	1	6	16	0.043	0.143	0.059	0.083
Gemini	1	9	16	0.038	0.1	0.059	0.074
	1	10	16	0.037	0.091	0.059	0.071
	1	9	16	0.038	0.1	0.059	0.074
	1	7	16	0.042	0.125	0.059	0.08
LLaMa	1	5	16	0.045	0.167	0.059	0.087
	1	5	16	0.045	0.167	0.059	0.087
	1	5	16	0.045	0.167	0.059	0.087
	1	5	16	0.045	0.167	0.059	0.087
	1	5	16	0.045	0.167	0.059	0.087
GPT	1	5	16	0.045	0.167	0.059	0.087
	1	5	16	0.045	0.167	0.059	0.087
	1	5	16	0.045	0.167	0.059	0.087
	1	6	16	0.043	0.143	0.059	0.083
	1	5	16	0.045	0.167	0.059	0.087
Mistral	1	7	16	0.042	0.125	0.059	0.08
	1	5	16	0.045	0.167	0.059	0.087
	1	7	16	0.042	0.125	0.059	0.08
	0	3	17	0	0	0	0
	1	7	16	0.042	0.125	0.059	0.08

TABLE LIX
ASSET PREDICTION FOR WHISPERGATE INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	0	11	15	0	0	0	0
	0	13	15	0	0	0	0
	0	13	14	0	0	0	0
	0	13	14	0	0	0	0
	0	13	15	0	0	0	0
Gemini	0	24	15	0	0	0	0
	0	20	15	0	0	0	0
	0	28	15	0	0	0	0
	0	27	14	0	0	0	0
LLaMa	0	7	15	0	0	0	0
	0	7	15	0	0	0	0
	0	7	15	0	0	0	0
	0	7	15	0	0	0	0
	0	7	15	0	0	0	0
GPT	0	8	19	0	0	0	0
	0	8	15	0	0	0	0
	0	7	15	0	0	0	0
	0	8	15	0	0	0	0
	0	8	15	0	0	0	0
Mistral	0	8	15	0	0	0	0
	0	7	15	0	0	0	0
	0	8	15	0	0	0	0
	0	7	15	0	0	0	0
	0	8	15	0	0	0	0

TABLE LX
RELATIONSHIP PREDICTION FOR WHISPERGATE INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	1	16	24	0.024	0.059	0.04	0.048
	1	14	24	0.026	0.067	0.04	0.05
	1	16	24	0.024	0.059	0.04	0.048
	1	13	24	0.026	0.071	0.04	0.051
	1	13	24	0.026	0.071	0.04	0.051
Gemini	0	21	24	0	0	0	0
	1	23	24	0.021	0.042	0.04	0.041
	1	24	24	0.02	0.04	0.04	0.04
	1	23	24	0.021	0.042	0.04	0.041
LLaMa	1	9	24	0.029	0.1	0.04	0.057
	1	9	24	0.029	0.1	0.04	0.057
	1	9	24	0.029	0.1	0.04	0.057
	1	9	24	0.029	0.1	0.04	0.057
	1	9	24	0.029	0.1	0.04	0.057
GPT	1	11	24	0.028	0.083	0.04	0.054
	1	11	24	0.028	0.083	0.04	0.054
	1	10	24	0.029	0.091	0.04	0.056
	1	12	24	0.027	0.077	0.04	0.053
	1	11	24	0.028	0.083	0.04	0.054
Mistral	1	13	25	0.026	0.071	0.038	0.05
	1	9	24	0.029	0.1	0.04	0.057
	1	13	25	0.026	0.071	0.038	0.05
	0	5	26	0	0	0	0
	1	13	25	0.026	0.071	0.038	0.05

TABLE LXI
ACTION PREDICTION FOR SWIFT INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	3	4	3	0.3	0.429	0.5	0.462
	2	5	4	0.182	0.286	0.333	0.308
	3	4	3	0.3	0.429	0.5	0.462
	3	4	3	0.3	0.429	0.5	0.462
	3	3	3	0.333	0.5	0.5	0.5
Gemini	2	4	4	0.2	0.333	0.333	0.333
	1	5	5	0.091	0.167	0.167	0.167
	2	3	4	0.222	0.4	0.333	0.364
	1	5	5	0.091	0.167	0.167	0.167
LLaMa	1	5	5	0.091	0.167	0.167	0.167
	1	5	5	0.091	0.167	0.167	0.167
	1	5	5	0.091	0.167	0.167	0.167
	1	5	5	0.091	0.167	0.167	0.167
	2	3	4	0.222	0.4	0.333	0.364
GPT	1	7	5	0.077	0.125	0.167	0.143
	1	6	5	0.083	0.143	0.167	0.154
	2	4	4	0.2	0.333	0.333	0.333
	1	8	5	0.071	0.111	0.167	0.133
	1	6	5	0.083	0.143	0.167	0.154
Mistral	1	3	5	0.111	0.25	0.167	0.2
	2	3	4	0.222	0.4	0.333	0.364
	1	3	5	0.111	0.25	0.167	0.2

TABLE LXII
ASSET PREDICTION FOR SWIFT INCIDENT

Model	TP	FP	FN	Accuracy	Precision	Recall	F1 Score
DeepSeek	0	8	4	0	0	0	0
	0	9	4	0	0	0	0
	0	10	3	0	0	0	0
	0	9	4	0	0	0	0
	0	9	4	0	0	0	0
Gemini	0	15	3	0	0	0	0
	0	12	3	0	0	0	0
	0	21	3	0	0	0	0
	0	12	4	0	0	0	0
	0	17	3	0	0	0	0
LLaMa	0	7	4	0	0	0	0
	0	7	4	0	0	0	0
	0	7	4	0	0	0	0
	0	7	4	0	0	0	0
GPT	0	10	4	0	0	0	0
	3	4	1	0.375	0.429	0.75	0.545
	0	7	4	0	0	0	0
	0	9	4	0	0	0	0
	0	7	1	0	0	0	0
Mistral	0	6	4	0	0	0	0
	0	6	4	0	0	0	0
	0	5	4	0	0	0	0

TABLE LXIII
RELATIONSHIP PREDICTION FOR SWIFT INCIDENT

Model	**TP**	**FP**	**FN**	**Accuracy**	**Precision**	**Recall**	**F1 Score**
DeepSeek	2	8	3	0.154	0.2	0.4	0.267
	2	10	4	0.125	0.167	0.333	0.222
	2	9	3	0.143	0.182	0.4	0.25
	3	9	3	0.2	0.25	0.5	0.333
	3	8	3	0.214	0.273	0.5	0.353
Gemini	2	9	4	0.133	0.182	0.333	0.235
	1	8	5	0.071	0.111	0.167	0.133
	2	8	4	0.143	0.2	0.333	0.25
	1	10	5	0.062	0.091	0.167	0.118
	1	8	4	0.077	0.111	0.2	0.143
LLaMa	1	9	5	0.067	0.1	0.167	0.125
	1	9	5	0.067	0.1	0.167	0.125
	1	9	5	0.067	0.1	0.167	0.125
	1	9	5	0.067	0.1	0.167	0.125
	1	9	5	0.067	0.1	0.167	0.125
GPT	1	15	5	0.048	0.062	0.167	0.091
	2	11	5	0.111	0.154	0.286	0.2
	2	8	4	0.143	0.2	0.333	0.25
	1	15	5	0.048	0.062	0.167	0.091
	1	11	5	0.059	0.083	0.167	0.111
Mistral	1	5	5	0.091	0.167	0.167	0.167
	2	5	4	0.182	0.286	0.333	0.308
	1	5	5	0.091	0.167	0.167	0.167