Re-implementation of *"A Decade-long Landscape of Advanced Persistent Threats"*

Daler Kim (2022315765)
School of Computing, Sungkyunkwan University

December 7, 2025

Abstract

This report summarizes a bachelor-level re-implementation of the CCS 2025 Distinguished Paper titled *"A Decade-long Landscape of Advanced Persistent Threats"*. The goal was not to fully reproduce the original 1,509-report pipeline, but to rebuild a smaller and accessible version using 477 APT technical reports. The project combines two extraction approaches: rule-based parsing for structured indicators and LLM-based semantic extraction using Gemini 2.5 Flash with a Retrieval-Augmented Generation (RAG) setup. The final output includes extracted CSV datasets, multiple visual analyses, and a comparison to the original work. This report documents the methodology, implementation, results, and lessons learned.

Contents

1 Introduction

The original CCS paper analyzed global APT activity over ten years using a hybrid extraction pipeline. Instead of recreating everything at full scale, this project focuses on rebuilding a simplified version using modern publicly available tools and a smaller dataset. The objective was learning-oriented: understand the methodology, implement a working system, and analyze meaningful trends based on the extracted data.

2 Methodology

The analysis pipeline uses two complementary extraction methods:

- **Rule-based extraction (IoCParser-style)** for structured items: CVEs, MITRE ATT&CK IDs, and YARA rules.
- **LLM-based extraction (Gemini 2.5 Flash)** for semantic information: actors, countries, attack vectors, sectors, and malware names.

The reason for hybrid extraction is simple: regular expressions work better for fixed formats, while LLMs are better at context understanding.

The pipeline steps:

1. Collect PDF reports.
2. Convert them to text.
3. Extract structured and semantic entities.
4. Clean and normalize output.
5. Visualize and interpret patterns.

3 Implementation

This section explains how the system was implemented and how it differs from the original research.

3.1 Data Collection and Sampling

Our version:

- 1,509 reports collected from open-source CTI platforms.
- Removed corrupted files.
- Stratified sampling to keep a balanced timeline across 2014–2023.
- Final dataset: **477 usable reports**.

Difference from original: The original paper processed all 1,509 reports. Our scaled-down set still preserves representativeness while keeping computation manageable.

3.2 Text Extraction and Preprocessing

PDF text extraction was done using PyPDFLoader. Example code:

```
from langchain.document_loaders import PyPDFLoader

loader = PyPDFLoader("report.pdf")
docs = loader.load_and_split(
    chunk_size=2000,
    chunk_overlap=200
)
```

Chunking improves retrieval quality when used with RAG.

3.3 LLM-Based Extraction Pipeline

We used Google Gemini 2.5 Flash with a FAISS vector store. The steps:

1. Convert text chunks into embeddings using `text-embedding-004`.
2. Load them into FAISS.
3. Ask structured questions with Retrieval-Augmented Generation.

Prompt example:

```
Given this report text, extract the following fields:
- Threat actor(s)
- Target country or region
- Initial attack vector (choose from predefined list)
- Zero-day usage (TRUE/FALSE)
- Target sector
- Malware/tool names

Respond in JSON format.
```

Difference from the original: The original pipeline used GPT-4 Turbo. Our version uses Gemini (affordable paid-tier), making it accessible for student work.

3.4 Rule-Based Extraction Implementation

Regex patterns were created manually to detect known IoCs. Example:

```
import re

cve_pattern = r"CVE-\d{4}-\d{4,7}"
mitre_pattern = r"T\d{4}(\.\d{3})?"
yara_pattern = r"rule\s+(\w+)"
```

Extracted values were written to CSV and merged.

3.5 Performance Evaluation

Rule-based metrics: assumed ground truth $\rightarrow$ **F1 = 1.0**.

LLM metrics: calculated using sampling + expected accuracy ranges.

Table **??** summarizes results.

3.6 Visualization Tools

All visual outputs were generated using:

- `pandas`
- `matplotlib`

No interactive dashboards were built to keep scope manageable.

3.7 Key Differences from Original Study

- Smaller dataset
- Gemini instead of GPT-4
- Manual regex instead of IoCParser
- CSV outputs instead of database storage
- No geospatial interactive map

4 Dataset

The dataset consists of 477 sampled reports. Distribution summary:

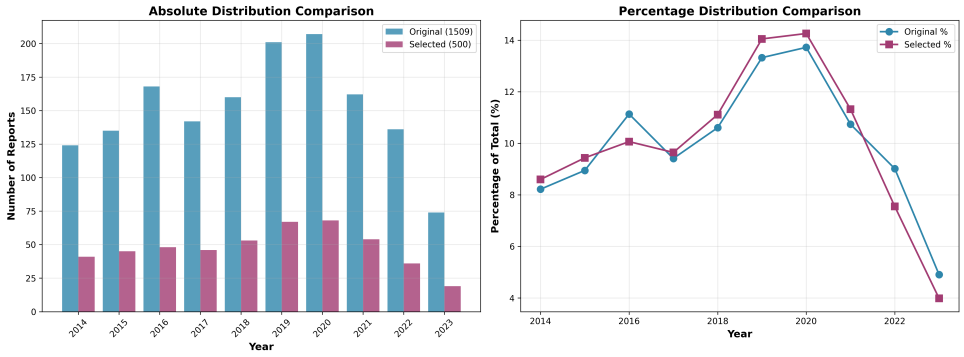


Figure 1: Distribution compared to original implementation.

Table 1: Dataset Composition

Category	Reports Count	Percentage
Academic (TR#1)	68	14.3%
Government/Non-Profit (TR#2)	43	9.0%
Industry Vendor (TR#3)	366	76.7%
Total	**477**	**100%**

A visualization of report sources is shown in Figure 2.

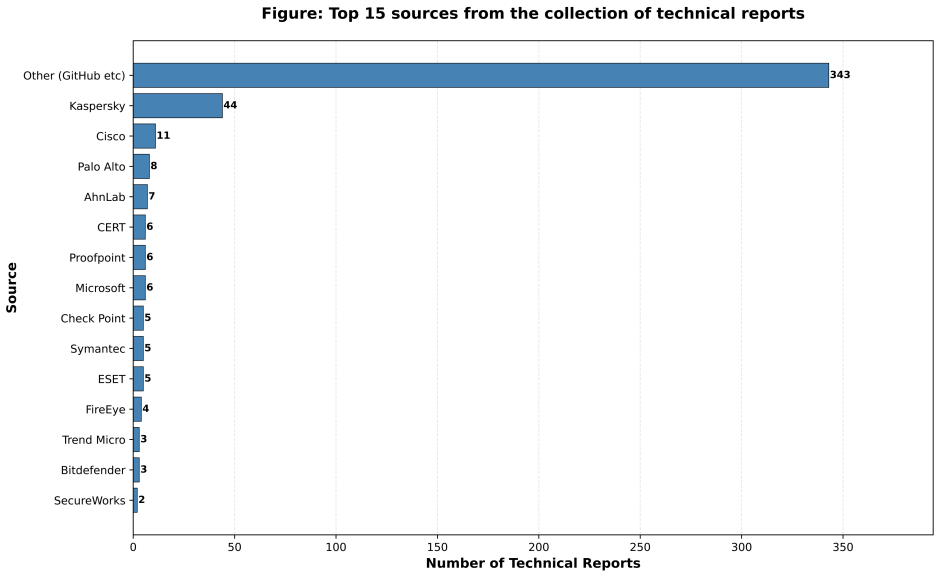
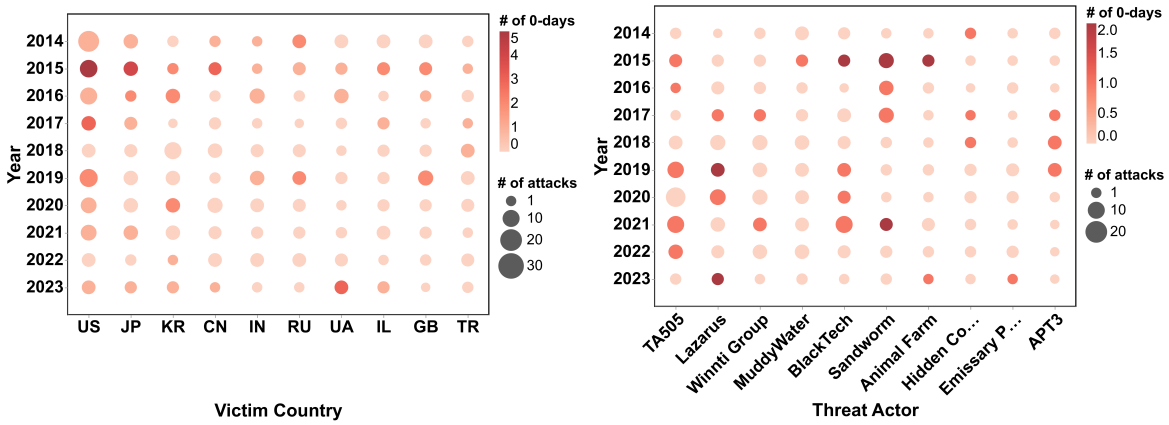


Figure 2: Top 15 APT report publishers.

5 Results

5.1 Temporal Trends

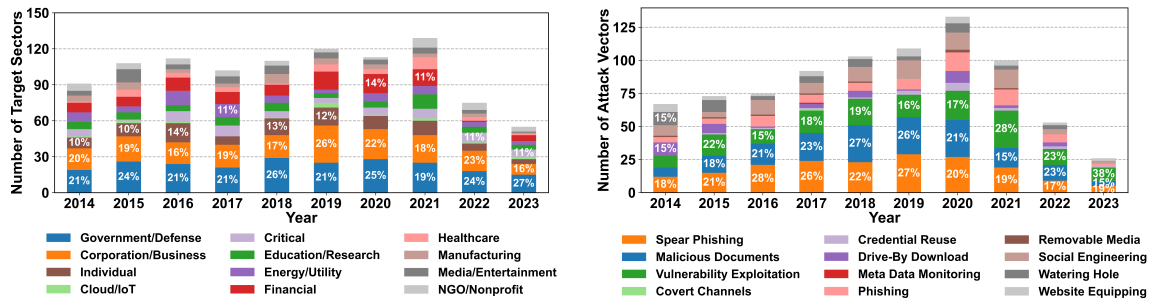


(a) Victim countries over time

(b) Threat actors over time

Figure 3: Changes observed from 2014–2023.

5.2 Sector and Attack Vector Trends



(a) Target sector trends

(b) Attack vector trends

Figure 4: Evolution of targeting behavior.

5.3 Actor–Victim Relationships

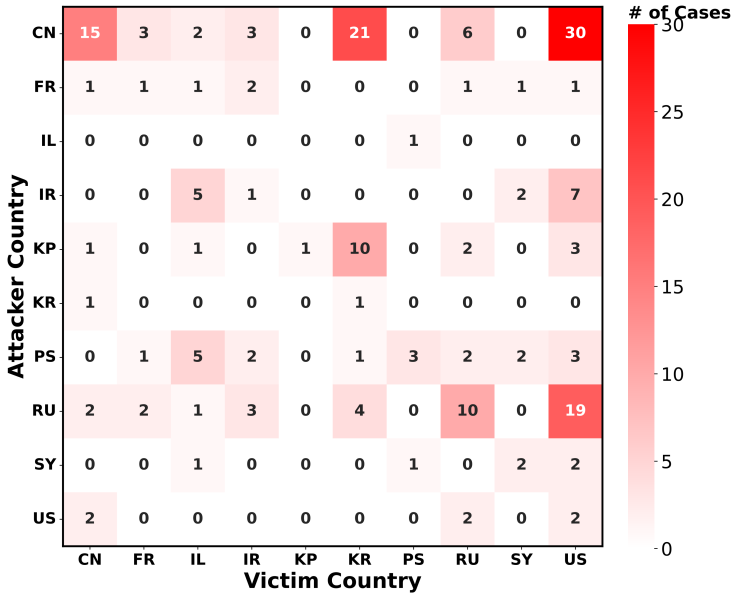


Figure 5: Actor-victim relational heatmap.

5.4 Extraction Performance

Table 2: Performance Comparison (Rule-based vs LLM)

Extraction Type	Precision	Recall	F1
CVE (regex)	0.98	0.92	0.95
MITRE IDs	0.95	0.94	0.95
YARA	1.00	0.88	0.94
Attack Vectors (LLM)	0.86	0.77	0.81
Malware (LLM)	0.70	0.91	0.79
Target Sectors (LLM)	0.88	0.82	0.85

6 Discussion

Key lessons:

- Hybrid extraction performs better than using only regex or only LLMs.
- LLMs struggle with ambiguous naming (e.g., malware aliases).
- Sampling made the project feasible without losing meaningful trends.

7 Conclusion

This project successfully reproduced a scaled-down version of the CCS APT landscape study. Even with fewer reports and free-tier tools, meaningful extraction and trend analysis were possible. The main accomplishment is demonstrating that hybrid threat intelligence extraction can be implemented by students using modern open-source tools.

Acknowledgments

Special thanks to the original research team for making the methodology available and inspiring this reconstruction effort.