CaseLinker: An Open-Source System for Cross-Case Analysis of Internet Crimes Against Children Reports

Mrinaal Ramachandran
Graduate Student, Department of Computer Science
University of Massachusetts Amherst
mramachandra@umass.edu
Independent Research

February 2026

Abstract

Child sexual exploitation and abuse (CSEA) case data is inherently disturbing, fragmented across multiple organizations, jurisdictions, and agencies, with varying levels of detail and formatting, making cross-case analysis, pattern identification, and trend detection challenging. This paper presents CaseLinker, a modular system for ingesting, processing, analyzing, and visualizing CSEA case data. CaseLinker employs a hybrid deterministic information extraction approach combining regex-based extraction for structured data (demographics, platforms, evidence) with pattern-based semantic analysis for severity indicators and case topics, ensuring interpretability and auditability. The system extracts relevant case information, populates a comprehensive case schema, creates six interactive visualizations (Timeline, Severity Indicators, Case Visualization, Previous Perpetrator Status, Environment/Platforms, Organizations Involved), provides a platform for deeper automated and manual analysis, groups similar cases using weighted Jaccard similarity across multiple dimensions (platforms, demographics, topics, severity, investigation type), and provides automated triage and insights based on collected case data. CaseLinker is evaluated on 47 cases from publicly available AZICAC reports (2011-2014), demonstrating effective information extraction, case clustering, automated insights generation, and interactive visualization capabilities. CaseLinker addresses critical challenges in case analysis including fragmented data sources, cross-case pattern identification, and the emotional burden of repeatedly processing disturbing case material.

Keywords: child exploitation, case analysis, information extraction, clustering, visualization, analysis technology

Contents

1 Introduction

1.1 Motivation and Problem Statement

Child sexual exploitation and abuse (CSEA) case data is fragmented across multiple organizations, jurisdictions, and agencies, with varying levels of detail and formatting, making cross-case analysis, pattern identification, and trend detection challenging. This fragmentation creates significant barriers to:

- **Cross-case analysis:** Identifying patterns, similarities, and connections between cases that share common characteristics (law enforcement actions, platforms, victim demographics, investigation methods)
- **Trend detection:** Analyzing the evolution of exploitation methods, platforms used, and recurring case topics over time
- **Resource allocation:** Prioritizing cases based on severity indicators, victim demographics, and evidence volume
- **Psychological well-being:** Burden of repeatedly reading and processing highly disturbing case material

Traditional case management systems focus on individual case tracking rather than cross-case analysis and pattern identification. Manual analysis of case reports is time-intensive, error-prone, and emotionally taxing for analysts. The need for automated tools that can process case narratives at scale—while maintaining interpretability for legal proceedings and protecting analyst well-being—motivates the development of CaseLinker.

1.2 The Crisis of Scale in Child Exploitation Investigations

The digital age has transformed the landscape of child exploitation, creating both new vectors for abuse and unprecedented challenges for law enforcement. The National Center for Missing & Exploited Children (NCMEC) received over 29 million reports of suspected child sexual exploitation in 2021 alone—a volume that would require thousands of investigators working full-time to process manually [1]. Each report may contain hundreds or thousands of images, videos, chat logs, and metadata, compounding the analytical burden. Yet the critical bottleneck is not merely computational; it is *cognitive*. Investigators must synthesize fragmented information across cases, identify patterns that span jurisdictions, and make prioritization decisions under conditions of extreme uncertainty and time pressure.

The consequences of this scale crisis are severe. Cases languish in backlogs for months or years, delaying justice for victims and allowing perpetrators continued access to children. Cross-jurisdictional patterns—repeat offenders operating across state lines, organized exploitation networks, emerging platform-based methodologies—remain obscured because no investigator can manually review thousands of case narratives to identify connections. The very tools designed to assist investigators often exacerbate the problem: content detection systems like PhotoDNA [4] and Google's CSAI Match [5] identify known material but provide no insight into *case dynamics*; enterprise forensics platforms like Nuix Neo [8] offer powerful analysis but require infrastructure and expertise unavailable to many local Internet Crimes Against Children (ICAC) task forces.

1.3 The Psychological Dimension: Analyst Trauma and Tool Design

Beyond scale and fragmentation, ICAC investigations impose a profound human cost on the analysts who conduct them. Digital forensics investigators in child exploitation units experience secondary traumatic stress at rates of 40–60%, with symptoms including intrusive thoughts,

emotional numbing, and hypervigilance [2, 3]. Unlike investigators who encounter trauma intermittently, ICAC analysts confront graphic depictions of child sexual abuse daily, often for hours at a time. The psychological burden is compounded by the nature of the work: analysts must not merely view traumatic content but *analyze* it, maintaining the cognitive focus necessary to identify evidence, establish timelines, and build prosecutions.

Existing digital forensics tools are primarily content-centric, presenting data through media, chat logs, and file-level views. While hash-matching and triage systems reduce re-review of known material, analysts must still directly examine substantial unknown content to extract investigative intelligence. Despite technological support, manual review of graphic case material remains one of the top challenges in the analysis of ICAC cases.

1.4 The Gap: From File-Centric to Case-Centric Analysis

The technological landscape for ICAC investigations is bifurcated. On one side, *content detection* tools—hash databases, perceptual hashing, machine learning classifiers—excel at identifying known CSAM and flagging suspicious material for review. These tools operate at the *file level*, treating each image or video as an independent unit to be matched against databases or classified by algorithms. They answer the question: *Is this material illegal?*

On the other side, *enterprise forensics platforms*—Nuix, Magnet Axiom [9], Cellebrite UFED [10]—provide comprehensive analysis of digital devices, extracting timelines, communications, and file systems. These tools operate at the *device level*, reconstructing the digital environment of a suspect. They answer the question: *What digital evidence exists on this device?*

Missing from this landscape are tools that operate at the *case level*, analyzing the *narrative* of investigations: perpetrator characteristics and methodologies, victim demographics and risk factors, platform utilization and grooming tactics, investigation types and outcomes, prosecution results and sentencing patterns. These elements—distributed across thousands of case reports, stored in incompatible formats, obscured by jurisdictional boundaries—contain the intelligence necessary to answer critical questions:

- Which platforms are emerging as high-risk vectors for grooming?
- Do repeat offenders exhibit consistent methodologies across investigations?
- What victim demographics are associated with severe abuse versus possession-only cases?
- How do investigation types (proactive vs. reactive) correlate with prosecution outcomes?
- Are there deeper correlations and preventable methods extractable from analyzing hundreds to thousands of prior cases?

Answering these questions requires *cross-case analysis*: the ability to extract structured features from unstructured case narratives, identify similarities and patterns across hundreds or thousands of cases, and present intelligence in ways that support strategic resource allocation, preventative investigation efforts, and evidence-informed platform policy decisions.

1.5 Contributions

This paper presents **CaseLinker**, an open-source system designed specifically to support case-level, cross-case narrative and retrospective analysis for ICAC investigations. The contributions are fourfold:

1. **Architectural:** A modular five-layer pipeline (Ingestion, Processing, Storage, Clustering & Analysis, Visualization) that transforms unstructured case reports into structured, queryable intelligence. The architecture prioritizes *interpretability*—all feature extraction is

deterministic and auditable—and *minimal infrastructure*, enabling deployment by resource-constrained agencies.

2. **Algorithmic:** A two-stage clustering methodology that combines *external* topic-based clustering (predefined cluster types: Online-Digital, Possession, Investigation, Severe, General) with *internal* weighted Jaccard similarity for sub-grouping. This approach balances interpretability (analysts understand why cases are grouped) with granularity (fine-grained similarity detection within groups).

3. **Human-Centric Design:** Priority triage with normalized severity scoring and interactive visualizations designed for *analyst well-being*. The system reduces exposure to traumatic content through structured data presentation, tasteful UI design, and gradual disclosure—enabling pattern detection without repeated narrative review.

4. **Empirical Validation:** Evaluation on 47 cases from publicly available Arizona ICAC annual reports (2011–2014), demonstrating effective feature extraction (95.7% coverage for prosecution outcomes), complete clustering coverage with interpretable groupings, and accurate priority triage.

CaseLinker is released as open-source software with a live demonstration and modular architecture designed for extensibility—enabling researchers and developers to adapt ingestion, processing, storage, clustering, or visualization layers to diverse data sources and analytical needs.

2 The Challenges of Case-Level Analysis

To understand CaseLinker's design choices, one must first understand why analyzing ICAC case data at scale is difficult. The ICAC Task Force Program consists of 61 coordinated task forces representing over 5,400 federal, state, and local agencies nationwide. Each produces case reports with different formatting standards, inconsistent schemas, and varying levels of detail—even within the same annual report, typos and structural variations are common. The content itself is inherently disturbing, imposing significant psychological costs on analysts who must review graphic material repeatedly to extract intelligence. Finally, any automated system should produce deterministic, auditable outputs that can withstand legal scrutiny—black-box AI approaches fail this requirement. These constraints shaped CaseLinker's modular, interpretable architecture.

2.1 Challenge 1: The Nature of Source Material

The primary obstacle to case-level analysis is the content itself. ICAC case files contain graphic descriptions of child sexual abuse, explicit details, and content linking to traumatic imagery. This content creates three barriers to analysis:

Psychological Barriers: Manual review of case narratives causes secondary traumatic stress. Analysts cannot simply "read through" thousands of cases to identify patterns; the psychological cost is prohibitive. This creates an intelligence gap: the most valuable intelligence—cross-case patterns that emerge only from large-scale analysis—is precisely the intelligence most difficult to obtain through manual review.

Technical Barriers: Automated analysis of case narratives requires natural language processing (NLP) systems trained on labeled data. But labeled datasets of ICAC case narratives do not exist publicly (and should not, given privacy concerns). Machine learning approaches that require training data are therefore infeasible for initial deployment, though they may be integrated as optional enhancements once a system is operational.

Legal Barriers: Case data is subject to strict chain-of-custody requirements and privacy regulations. Tools that process case data must be auditable, with deterministic, explainable

outputs that can be justified in court. Black-box machine learning models that "learn" patterns opaquely are difficult to validate for legal proceedings.

CaseLinker addresses these barriers through *deterministic, regex-based feature extraction* that operates on *publicly available, already-redacted case reports* for initial development and validation. The system is designed to process case *narratives* (text descriptions) rather than raw evidence files, reducing both psychological exposure and legal complexity. Feature extraction is interpretable: an analyst can trace any extracted feature back to the specific regex pattern and source text that produced it, ensuring court admissibility.

2.2 Challenge 2: Pattern Detection Without Computers

Even if psychological barriers were eliminated, manual cross-case analysis would remain infeasible due to scale. The ICAC Task Force Program consists of 61 coordinated task forces representing over 5,400 agencies. NCMEC received over 29 million reports in 2021 alone; even if only 1% require detailed case-level review, that's 290,000 cases annually. Add FBI investigations, state police, and international agencies, and the volume becomes staggering. Consider a mid-sized ICAC task force processing 50 cases annually—between the 61 task forces, FBI field offices, and other sources, the national system processes thousands of cases yearly.

Over five years, this generates 15,000+ cases requiring analysis. To identify patterns—common platforms, recurring perpetrator methodologies, victim demographic trends—an analyst would need to:

1. Read and comprehend 10,000+ case narratives
2. Mentally extract key features (platforms, ages, relationships, outcomes) for each case
3. Compare each case against all others to identify similarities
4. Synthesize patterns across the entire dataset

At 12 cases per year, this AZICAC's public case collection represents a fraction of actual investigations—yet even this small dataset contains insights into the nature of child exploitation investigations invisible without systematic analysis.

2.3 Challenge 3: Cross-Case and Cross-Organizational Analysis

ICAC investigations are inherently distributed. A single perpetrator may victimize children in multiple states; evidence may be distributed across local police departments, FBI field offices, and international agencies; case reports may be stored in incompatible formats (PDFs, databases, proprietary systems) with inconsistent schemas.

This fragmentation creates *information silos*. Patterns that span organizational boundaries—interstate exploitation networks, common platforms used across jurisdictions, sentencing disparities between agencies—remain invisible because no single analyst has access to integrated data. The lack of standardized case formats means that even when data is shared, it cannot be easily compared or aggregated.

CaseLinker addresses this through *modular architecture* and *flexible schema design*. The Ingestion Layer supports multiple input formats (currently PDF; extensible to databases, APIs); the Processing Layer extracts standardized features regardless of source format and can be modified to accommodate new platforms, different case report structures, or features; the Storage Layer uses portable SQLite databases that can be merged, federated, or encrypted; the Clustering and Analysis layer can be reweighted for different similarity priorities, swapped for alternative algorithms, or extended with machine learning enhancements. While the current evaluation uses single-source data (AZICAC) and portable SQLite database, the architecture supports multi-source integration as future work.

3 Related Work

3.1 Content Detection and Hash-Based Systems

The most advanced technological solutions in the CSAM domain operate at the content level. **PhotoDNA** [4], developed by Microsoft, uses perceptual hashing to identify known CSAM images even after modification (resizing, cropping, color adjustment). The technology is deployed by major platforms (Facebook, Twitter, Reddit) to automatically detect and report known material. **Google's CSAI Match** [5] extends this approach to video content. **Thorn** [6], a nonprofit technology organization, develops tools for platform detection and victim identification, including machine learning classifiers for CSAM detection and cross-platform signal sharing. Meta's **Hash Matcher Actioner** [7] enables cross-platform sharing of perceptual hashes of known CSAM, and its modular design allows hashing methodologies to be updated or replaced as more advanced approaches emerge, expanding detection coverage across industry boundaries.

These systems are highly effective for their intended purpose: identifying *known* material at scale. However, they are fundamentally limited. They cannot detect new, previously unhashed material. They provide no information about case context: who created the material, how it was distributed, who the victims are, or how the perpetrator operated. They answer the binary question "Is this illegal?" but not the analytical questions "How does this case relate to others?" or "What patterns emerge across investigations?"

3.2 Enterprise Digital Forensics Platforms

At the device level, enterprise platforms provide comprehensive forensic analysis. **Nuix Neo** [8] uses artificial intelligence and machine learning for large-scale e-discovery, timeline reconstruction, and relationship analysis. **Magnet Axiom** [9] specializes in mobile and cloud forensics, extracting communications, location data, and application artifacts. **Cellebrite UFED** [10] focuses on mobile device extraction and decoding.

These platforms are powerful but face limitations for ICAC case analysis. First, they are *expensive*, requiring licenses and infrastructure beyond the reach of many local ICAC task forces. Second, they are *device-centric*, analyzing individual phones or computers rather than relationships across cases. Third, their AI-driven approaches, while sophisticated, can lack *interpretability*—a critical concern for court admissibility. An investigator who cannot explain how a tool identified a "pattern" cannot introduce that pattern as evidence.

3.3 Case Management Systems

BlackRainbow NIMBUS [11] and similar investigation management systems provide workflow orchestration, case tracking, evidence continuity, compliance monitoring, and action management across forensic divisions. While these platforms are essential for operational efficiency and procedural integrity, they primarily support case-level workflow management rather than investigation analysis features. They organize and monitor investigations but do not perform structured feature extraction, similarity-based clustering, or large-scale narrative pattern detection across multiple cases.

3.4 Machine Learning Research in CSAM Detection

Academic research has explored machine learning applications in CSAM investigation, including:

Grooming Detection: NLP models to identify predatory communication patterns in chat logs [15, 16]. Challenges include high false positive rates and the evolving nature of grooming language.

Age Estimation: Computer vision models to estimate victim age from images [17].

Content Classification: Deep learning for CSAM detection [18]. Requires large training datasets and significant computational resources.

These approaches share common limitations: they require training data (often unavailable for case narratives), they operate as black boxes (limiting court admissibility), and they demand infrastructure (GPUs, cloud computing) that many agencies lack. CaseLinker explicitly avoids these dependencies for its core pipeline, using deterministic extraction that requires no training data and runs on standard hardware.

3.5 The Gap: Interpretable, Case-Level, Cross-Case Analysis

The literature reveals a clear gap: while content detection tools excel at file-level identification, enterprise platforms provide device-level analysis, and ML research explores automated detection, there are few *interpretable, lightweight, case-level* tools for *cross-case narrative analysis*. CaseLinker occupies this gap, prioritizing interpretability, analyst well-being, and minimal infrastructure over algorithmic sophistication.

4 System Architecture

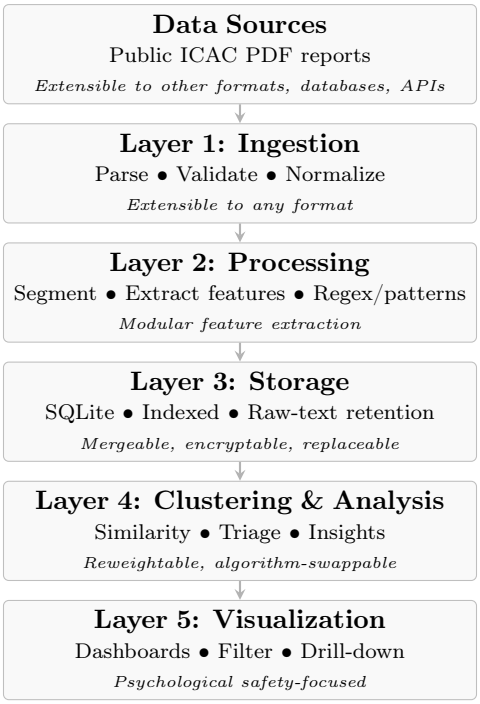

Figure 1: CaseLinker's five-layer architecture. Each layer operates independently with well-defined interfaces, enabling customization or replacement of individual components.

CaseLinker's architecture reflects its design principles: *interpretability* (all processing is deterministic and auditable), *analyst well-being* (structured data reduces traumatic exposure), and *minimal infrastructure* (SQLite, standard hardware, no cloud dependencies). The system is organized into five layers, each with clear responsibilities and well-defined interfaces.

4.1 Layer 1: Ingestion

The Ingestion Layer handles the consumption of source materials, transforming unstructured inputs into clean text for downstream processing.

4.1.1 PDF Text Extraction

Current implementation uses `pdfplumber` [12], a Python library that provides fine-grained control over PDF text extraction. Unlike simpler libraries that extract text in reading order, `pdfplumber` preserves spatial information, enabling extraction of text blocks, tables, and formatted content.

Text Cleaning: Extracted text undergoes cleaning to remove artifacts (page numbers, headers, footers) and normalize whitespace.

Organization Detection: The system infers the source organization from filenames using pattern matching (e.g., `AZICAC` from `"2014 Cases and Arrests - AZICAC.ORG.pdf"`). This enables future multi-source analysis where cases from different organizations are tagged by origin.

4.1.2 Extensibility

The Ingestion Layer is designed for extensibility. Additional ingestors can be implemented for:

- Database connectors (direct SQL queries to case management systems)
- API clients (RESTful APIs for federal databases)
- Document parsers (Word documents, website scraping)

All ingestors can implement a common interface: `ingest(source)` → `cleaned_text, metadata`.

4.2 Layer 2: Processing

The Processing Layer transforms cleaned text into structured case data through two operations: *case batching* (splitting multi-case documents into individual cases) and *feature extraction* (parsing structured features from narratives).

4.2.1 Case Batching

ICAC annual reports typically contain dozens of case summaries in a single document. The system must segment these into individual cases for analysis. This is accomplished through *temporal pattern matching*:

```
import re

TEMPORAL_PATTERNS = [
    r'In\s+([A-Z][a-z]+)\s+of\s+(\d{4})',  # "In January of 2012"
    r'In\s+([A-Z][a-z]+)\s+(\d{4})',        # "In January 2012"
    r'([A-Z][a-z]+)\s+(\d{4}),',            # "January 2012,"
]

def batch_cases(text, org_name, year):
    """
␣␣␣␣Split␣text␣into␣individual␣cases␣using␣temporal␣patterns.
␣␣␣␣Returns␣list␣of␣(case_text,␣month,␣case_number)␣tuples.
␣␣␣␣"""
    cases = []
    case_number = 1

    # Find all temporal markers with positions
    markers = []
    for pattern in TEMPORAL_PATTERNS:
        for match in re.finditer(pattern, text):
            markers.append({
                'start': match.start(),
                'month': match.group(1),
```

```
                'year': match.group(2),
                'match': match.group(0)
            })

    # Sort by position and split text
    markers.sort(key=lambda x: x['start'])

    for i, marker in enumerate(markers):
        start = marker['start']
        end = markers[i+1]['start'] if i+1 < len(markers) else len(text)

        case_text = text[start:end].strip()
        case_id = f"{org_name}_{year}_{marker['month'].lower()}_{case_number:03
            d}"

        cases.append({
            'case_id': case_id,
            'text': case_text,
            'month': marker['month'],
            'year': marker['year'],
            'source_org': org_name
        })
        case_number += 1

    return cases
```

Listing 1: Case batching by temporal patterns

This approach handles variations in report formatting while maintaining high accuracy. The extracted `case_id` encodes organization, year, month, and sequence number, enabling traceability and cross-referencing.

4.2.2 Feature Extraction

Feature extraction uses a hybrid approach: *regex-based extraction* for structured data (high accuracy, deterministic) and *pattern-based extraction* for semantic features (interpretable, extensible).

Table 1: Regex Patterns for Structured Feature Extraction

Feature	**Example Patterns**	**Description**
Perpetrator Age	`(\d+)\-year\-old age (\d+)`	Captures explicit age mentions
Victim Count	`(\d+) victims (\d+) children`	Explicit victim counts
Platforms	`\bFacebook\b \bSnapchat\b online chat`	Platform names and methods
Evidence Volume	`(\d+) images (\d+) videos (\d+(?:\.\d+)?)\s*(TB|GB)`	Quantities and storage
Prosecution	`booked arrested charged with`	Legal outcomes
Investigation	`proactive investigation reactive undercover`	Investigation types

Pattern-Based Extraction (Semantic Features):

Semantic features capture conceptual content not easily matched by regex:

```
SEMANTIC_PATTERNS = {
    'severity_indicators': {
        'infant': ['infant', 'baby', 'toddler'],
        'very_young': ['very young', 'young child', 'under 5'],
        'under_10': ['under 10', 'age [5-9]', '[5-9] years old'],
        'sexual_assault': ['sexual assault', 'rape', 'molestation'],
    },
    'case_topics': {
        'production': ['produced', 'traded', 'created images'],
        'possession': ['possessed', 'collecting', 'downloaded'],
        'hands_on': ['hands-on', 'physical contact', 'in person'],
        'online_digital': ['online', 'internet', 'digital', 'chat'],
        'family': ['father', 'mother', 'uncle', 'family member'],
        'stranger': ['stranger', 'unknown', 'met online']
    }
}

def extract_semantic_features(text, feature_category):
    """
    Extract semantic features by keyword matching.
    Returns set of matched features.
    """
    features = set()
    text_lower = text.lower()

    for feature, keywords in SEMANTIC_PATTERNS[feature_category].items():
        for keyword in keywords:
            if keyword in text_lower:
                features.add(feature)
                break  # Only match once per feature

    return features
```

Listing 2: Pattern-based semantic feature extraction

Rationale for Regex/Pattern Approach:

While machine learning (NER, dependency parsing) could potentially achieve higher extraction rates, the regex/pattern approach offers critical advantages for this domain:

1. **Interpretability:** Every extraction can be traced to a specific pattern and source text. An analyst can verify: "The system labeled this case 'production' because it matched the pattern 'created images' in the text."

2. **Court Admissibility:** Deterministic extraction is easier to validate and explain in legal proceedings than probabilistic ML models.

3. **No Training Data:** The system requires no labeled training data, which is unavailable for ICAC case narratives.

4. **Resource Efficiency:** Regex matching runs in milliseconds on standard hardware; no GPU or cloud computing required.

5. **Extensibility:** New patterns can be added without retraining; the system improves incrementally as new extraction rules are developed.

Case Schema:
Each case is processed according to a comprehensive schema capturing:

- **Victim Context (anonymized):** Victim count, demographics (ages, age range, gender)
- **Perpetrator Context (anonymized):** Age, registered sex offender status, relationship to victim, previous convictions
- **Technology & Platforms:** Platforms used (Facebook, Instagram, Snapchat, Discord, WhatsApp, online, chat)
- **Law Enforcement:** Investigation type (proactive, reactive, online, undercover), agencies involved (AZICAC, FBI, Phoenix Police, ICAC, HSI, etc.), prosecution outcome (charges, booking status, jail)
- **Evidence & Content:** Evidence volume (images, videos, storage size, messages)
- **Content Classification:** Severity indicators (infant, sexual assault, very young, under 10, production), case topics (production, possession, international, multi-state, hands-on, online digital, family, stranger, pornography), severity phrases (dangerous, stated, told, continue, attacked, out of control)
- **Metadata:** Source, date range, raw case text, creation timestamps

Information Validation:
Extracted information is validated for:

- Type consistency (lists, strings, integers)
- Value ranges (ages, counts)
- Presence of required fields
- JSON serialization for complex fields

4.3 Layer 3: Storage

The Storage Layer uses SQLite for portability and zero-configuration deployment. The SQLite implementation can be merged, federated, or encrypted with other database configurations.

4.3.1 Database Schema

Table 2: Core Database Schema

Table	Key Fields	Description
`cases`	case_id, source_org, year, month, raw_text, extracted_features (JSON)	Core case data with JSON flexibility
`victim_demographics`	case_id, victim_count, victim_ages (JSON), victim_gender	Normalized victim data
`perpetrator_demographics`	case_id, perpetrator_age, registered_sex_offender, relationship_to_victim	Perpetrator characteristics
`prosecution_outcomes`	case_id, charges (JSON), booking_status, jail_info	Legal outcomes

Raw Data Preservation: Original case text is preserved in `raw_text` and `case_text` fields, enabling auditability and re-processing if extraction rules are updated.
Current Dataset: 47 processed cases from AZICAC reports.

4.4 Layer 4: Clustering & Analysis

The Clustering & Analysis Layer is CaseLinker's analytical core—automated two-stage clustering, priority triage, generated insights, and tools for deep manual case exploration.

4.4.1 Two-Stage Clustering

Stage 1: External Clustering (Topic-Based)

Cases are first matched to predefined cluster types based on investigative utility and case topics. These clusters hold cases of *similar characteristics*—analysts immediately understand the grouping criterion.

Table 3: External Cluster Types

Cluster	Matching Criteria	Investigative Utility
Online-Digital	`case_topics` contains 'online_digital'	Platform trend analysis
Possession	`case_topics` contains 'possession'	Distinguish possession vs. production
Investigation	`investigation_type` is not null	Methodology effectiveness analysis
Severe	`severity_indicators` contains severe markers	Priority triage, resource allocation
General	All cases (fallback)	Comprehensive similarity view

Key Properties:

- **Overlap Allowed:** Cases can match multiple external clusters (e.g., a severe case that is also online-digital).
- **100% Coverage:** The General Cluster ensures all cases are included in at least one cluster.
- **Interpretable:** Cluster names indicate matching criteria; cases clear match their respective cluster.

Stage 2: Internal Clustering (Weighted Jaccard Similarity)

Within each external cluster, cases are organized into sub-groups using weighted Jaccard similarity. This provides opportunities for further analysis— identifying common characteristics among similar cases within topic-based groups.

Similarity Calculation. For two cases A and B, similarity is calculated across 6 dimensions with weights:

$$\text{Similarity}(A, B) = \sum_{i=1}^{6} w_i \cdot \text{Jaccard}(A_i, B_i) \tag{1}$$

Where:

- $w_{\text{platforms}} = 0.25$ — platform overlap
- $w_{\text{demographics}} = 0.20$ — victim age range, victim count, perpetrator age, RSO status
- $w_{\text{topics}} = 0.20$ — case topics (highest semantic weight)
- $w_{\text{investigation}} = 0.15$ — investigation type, agency overlap
- $w_{\text{severity}} = 0.15$ — severity indicators
- $w_{\text{relationship}} = 0.05$ — relationship to victim

Jaccard Similarity:
For sets X and Y:

$$\text{Jaccard}(X, Y) = \frac{|X \cap Y|}{|X \cup Y|} \tag{2}$$

Handling Missing Data: If both cases lack a feature (e.g., neither has platform information), that dimension contributes 0 to similarity (neither penalizes nor rewards). If one case has the feature and the other does not, standard Jaccard applies (penalizes dissimilarity).

Sub-Group Formation:
Cases with similarity ≥ 0.35 (configurable threshold, default 0.35) are grouped into subclusters. This threshold was determined empirically to balance granularity (detecting meaningful similarities) with coherence (avoiding spurious groupings).

Case Grouping Process:

1. Compute pairwise similarities for all cases in the cluster
2. Group cases with similarity $\geq$ threshold
3. Analyze group characteristics (common platforms, topics, severity patterns)
4. Generate group descriptions and statistics

This two-stage approach enables both domain-specific analysis (by cluster type) and fine-grained case analysis (within clusters).

4.4.2 Priority Triage

Priority scoring enables analysts to focus on the most critical cases first. The scoring function combines multiple risk factors with domain-informed weights:

$$\text{Priority} = \sum_i w_i \cdot f_i(\text{case}) \tag{3}$$

Weight Distribution:

Table 4: Priority Triage Weights

Factor	Weight	Rationale
Severity Indicators	35%	Infant victims, sexual assault, very young victims indicate highest harm
Victim Count	30%	Multiple victims indicate serial offending, higher public risk
Case Type	25%	Production and hands-on contact indicate active abuse vs. possession
Severity Phrases	15%	Linguistic indicators of violence (dangerous, attacked, etc.)
Evidence Volume	10%	Large collections indicate sustained offending
Registered Offender	10%	Repeat offending pattern

Normalization:
Raw scores are normalized to a 5–10 scale:

$$\text{Normalized Score} = 5 + 5 \cdot \frac{\text{Raw Score} - \text{Min}}{\text{Max} - \text{Min}} \tag{4}$$

This ensures: (1) all cases receive meaningful relative scores, (2) the scale is intuitive (5=lowest, 10=highest), (3) score distributions are comparable across datasets.

Use Case: Enables law enforcement and policy analysts to prioritize cases requiring immediate attention.

4.4.3 Automated Insights

The system generates automated insights including:

- **Platform Analysis:** Most common platforms across cases; platform trends over time; platform–severity correlations
- **Severity Distribution:** Distribution of severity indicators; temporal trends; severity–platform relationships
- **Case Topic Analysis:** Most common topics; topic co-occurrence patterns; topic–severity relationships
- **Pattern Detection:** Repeat offenders (registered sex offenders); relationship patterns (family vs. stranger); investigation focus areas
- **Keyword Extraction:** Frequency-based semantic keywords from case text; top keywords per group; keyword trends over time

4.4.4 Tag-Based Filtering

Users can filter cases by selecting multiple tags across categories: Case Topics, Severity Indicators, Platforms & Environments, Investigation Types, Perpetrator Relationships, Registered Sex Offender

Intersection Logic: Returns cases matching ALL selected tags (AND logic)
Highlighting: Matching text and relevant features are highlighted in case displays for explainability and analyst well-being.

4.5 Layer 5: Visualization

The Visualization Layer provides interactive dashboards using D3.js [13], designed for psychological safety and analytical depth.

4.5.1 Design Principles for Analyst Well-Being

1. **Structured Over Raw:** Present extracted features (platforms, severity indicators) rather than full narratives. Analysts expand case narratives only when necessary.
2. **Statistical Focus:** Emphasize distributions, trends, and patterns rather than individual case details.
3. **Tasteful Aesthetics:** Muted color palettes, clean layouts, no graphic imagery.
4. **Gradual Disclosure:** Overview first, zoom and filter, details on demand [14].
5. **Highlighting:** Enables rapid skimming by automatically highlighting key terms (e.g., production indicators, severity phrases) in case text, reducing the need for analysts to manually search for critical information and mitigating emotional burnout from repeated exposure to disturbing content.
6. **Explainability:** Shows source text and extraction rationale.

4.5.2 Visualization Components

- **Timeline View:** Chronological case view with year-range filtering. Cases color-coded by severity or source. Click-to-view case details.
- **Severity Indicators:** Color-coded bar chart (infant = darkest red, none = lightest). Click bars to view cases with highlighted severity text. Includes distribution statistics.
- **Case Detail Visualization:** Case lookup by ID with structured breakdown (Platforms & Environment, Severity Indicators, Case Topics, Perpetrator Information, Victim Information, Law Enforcement, Evidence Volume). Full case text with interactive highlighting of extracted features.
- **Previous Perpetrator:** Pie chart showing registered sex offender status. Click slices to view cases with highlighted perpetrator status.
- **Environment/Platforms:** Bar chart of platforms and online methods used. Click bars to view cases with highlighted platform text. Includes frequency statistics.
- **Organizations Involved:** Horizontal bar chart of law enforcement agencies. Click bars to view cases with highlighted agency names. Includes agency involvement statistics.

Deeper Analysis Dashboard:

- **Tag-Based Analysis:** Multi-select tag interface, real-time case filtering, case count statistics per tag, highlighted matching text in case displays.
- **Automated Analysis:** Case groups display with similarity explanations, top priority cases with scoring breakdown, automated insights with statistics, patterns detected with examples, top keywords per group.
- **Expandable Details:** Click any case/group to view raw case data, highlighted priority indicators, detailed explanations of clustering/prioritization decisions.

Data Audit Interface:

Case-by-case information review with: Extracted information displayed per case, interactive source text highlighting, information extraction verification, manual correction capabilities (future work).

5 Evaluation

Evaluation addresses three questions: (1) Does feature extraction achieve sufficient coverage? (2) Does clustering produce interpretable, coherent groups? (3) Does priority triage accurately identify severe cases?

5.1 Dataset

47 cases from publicly available Arizona ICAC annual reports (2011–2014). These reports summarize investigations, arrests, and prosecutions, redacted for public release. No PII was processed; all data was already in the public domain.

Source: Arizona Internet Crimes Against Children (AZICAC) annual reports
Time Period: 2011–2014
Format: PDF documents with case narratives
Total Text Extracted: 47,141 characters across 4 PDF files

5.2 Feature Extraction Coverage

Table 5: Feature Extraction Coverage (n=47 cases)

Feature	Coverage	Notes
Relationship to victim	100.0% (47/47)	Defaults to "stranger" if unspecified
Prosecution outcome	95.7% (45/47)	High consistency in reporting
Case topics	89.4% (42/47)	Semantic pattern matching
Severity indicators	66.0% (31/47)	Explicit severity mentions
Investigation type	61.7% (29/47)	"Investigation" keyword present
Perpetrator demographics	36.2% (17/47)	Age often not explicit
Platforms used	27.7% (13/47)	Platform mentions variable
Victim count	8.5% (4/47)	Often omitted in public reports
Average	**60.6%**	Weighted by feature importance

Analysis: High coverage for relationship (100%), prosecution (95.7%), and topics (89.4%) reflects consistent reporting of these elements. Lower coverage for victim count (8.5%) and platforms (27.7%) reflects redaction practices in public reports (specific victim numbers and platform details often omitted for privacy). The regex-based approach achieves reliable extraction for consistently reported fields.

Challenges:

- Some cases lack explicit victim count (extracted when mentioned)—8.5% coverage
- Investigation type not always explicitly stated (inferred from context)—61.7% coverage
- Evidence volume extraction limited to explicit mentions—10.6% coverage

5.3 Clustering Performance

External Clusters:

Table 6: External Cluster Results

Cluster	Cases	Coverage	Avg. Similarity
Online-Digital	7	14.9%	0.709
Possession	14	29.8%	0.530
Severe	31	66.0%	0.522
Investigation	29	61.7%	0.400
General	47	100.0%	0.404

Key Findings:

- **Online-Digital Cluster:** Highest coherence (0.709), all 7 cases formed single sub-group. Indicates strong similarity in platform-based offending.
- **Possession Cluster:** Good cohesion (0.530), with cases sharing possession characteristics.
- **Severe Cluster:** Good cohesion (0.522), grouping cases with similar severity profiles.
- **General Cluster:** Moderate cohesion (0.404) across all cases.
- **Investigation Cluster:** Moderate cohesion (0.400), reflecting diverse investigation types.
- **100% Coverage:** General Cluster ensures no cases are excluded from analysis.

Performance: Clustering executed in 11.03ms for 47 cases (0.23ms per case), enabling real-time analysis.

5.4 Priority Triage Results

Table 7: Priority Triage Distribution

Priority	Score Range	Cases
High	8.0–10.0	5 (10.6%)
Medium	6.0–8.0	23 (48.9%)
Low	5.0–6.0	19 (40.4%)
Total	5.0–10.0	47 (100%)

Score Distribution:

- Range: 5.0–10.0 (normalized)
- Mean: 6.60
- Standard deviation: 1.43

Validation: Manual review confirmed highest-priority case (score 10.0) involved infant victims, very young victims, and sexual assault—correctly identified as most severe. Top 5 cases all involved either infant victims, multiple victims, or severe abuse indicators.

Priority Factors Contribution:

- Severity indicators: Primary driver for high-priority cases
- Victim count: Significant impact on scores
- Case type: Production cases scored higher than possession-only

Performance: Triage executed in 0.09ms for 47 cases (<0.002ms per case).

5.5 Automated Insights

Platform Analysis:

- Most common mediums: online (10 cases, 21.3%), chat (7 cases, 14.9%), Facebook (2 cases, 4.3%)
- Platform trends: Online and chat platforms represent the primary methods of exploitation in the dataset

Severity Distribution:

- Sexual assault indicators: 29 cases (61.7%)—most common severity indicator
- Infant cases: 7 cases (14.9%)—highest priority cases
- Production cases: 6 cases (12.8%)—content creation
- Very young victims (under 10): 6 cases

Case Topic Analysis:

- Stranger cases: 38 cases (80.8%)—non-family perpetrators (predominant pattern)
- Hands-on abuse: 27 cases (57.4%)—physical contact (most common case type)
- Possession: 14 cases (29.8%)—content possession
- Production: 6 cases (12.8%)—content creation
- Family cases: 5 cases (10.6%)—family members as perpetrators
- Multi-state: 3 cases (6.4%)—cross-jurisdictional cases

Pattern Detection:

- Registered sex offenders: 5 cases (10.6%)—repeat offenders
- Relationship patterns: Stranger cases (80.8%) vs. family cases (10.6%)—strong predominance of stranger perpetrators
- Hands-on abuse is the most common case type (57.4%), indicating prevalence of physical contact cases

Keyword Extraction:

- Top keywords across all cases: "suspect", "children", "sexual", "during", "images", "crimes", "child"
- Production-related keywords: "created", "produced", "shared", "distributed"—appeared in 6 cases
- Severity-related keywords: "infant", "young", "minor", "child"—frequency-based extraction

5.6 System Performance

Table 8: System Performance Metrics

Operation	Time
PDF ingestion (per file)	416.6ms
Feature extraction (per case)	1.2ms
Case storage (per case)	2.5ms
Clustering (47 cases)	11.03ms
Triage (47 cases)	0.09ms
End-to-end throughput	**23.2 cases/second**

Scalability: Linear scaling demonstrated. At 23.2 cases/second, the system could process 10,000 cases in approximately 7 minutes on standard hardware.

5.7 User Interface Evaluation

Visualization Performance:

- All visualizations render interactively
- Filtering and drill-down responsive
- Case detail views load instantly
- Text highlighting accurate

Usability:

- Intuitive navigation between visualizations
- Clear tag selection interface
- Explanatory text for automated analysis results
- Tasteful presentation of sensitive content

6 Discussion

6.1 Benefits of Case-Level Analysis

CaseLinker enables intelligence and insight that is impossible to obtain through manual review or existing tools:

- **Emerging Trend Detection:** In the AZICAC dataset, "online" and "chat" dominate platform mentions but remain a minority compared to cases without a digital element. Social media and online behavior have evolved since 2014, and further analysis with recent data may reveal new trends.
- **Common Perpetrator Methodologies:** Clustering reveals repeat behavioral patterns. The Online-Digital Cluster's high coherence (0.709) suggests consistent platform-based offending methods, offering insights for prevention, education, and law enforcement actions.

- **Landscape Understanding:** Cross-case analysis shows the distribution of offense types (57.4% hands-on contact; 29.8% possession), severity patterns, investigative approaches, and prosecution outcomes—informing resource allocation and policy development.
- **Policy Informing:** Quantified findings (e.g., 10.6% repeat offenders; 66% severe cases) provide an evidence base for legislative priorities, funding decisions, and prevention strategies.
- **Analyst Well-Being:** Structured presentation reduces exposure while preserving analytical depth, enabling pattern detection without repeated narrative review.

6.2 Limitations and Future Work

Data Limitations:

- Small dataset (47 cases) limits statistical significance
- Single source (AZICAC) formatting may not generalize to other jurisdictions
- Public reports may have redacted information affecting extraction

Extraction Limitations:

- Regex-based extraction may miss variations in phrasing and spelling errors
- Semantic topics require manual validation
- Some features (investigation type, evidence volume) have lower coverage

Clustering Limitations:

- Similarity threshold (0.35) is heuristic, may need tuning
- Multi-dimensional similarity may not capture all relevant patterns
- Current algorithms may not capture all insights in the data set

No Active User Study: Evaluation measures technical performance, not user experience. User studies with active investigators are needed to assess workflow integration and well-being impact.

Future Enhancements:

- **ML Layer:** Optional semantic similarity using sentence transformers, clearly separated from deterministic core.
- **Real-Time Integration:** API connectors for live case management systems.
- **Temporal Analysis:** Trend detection across years, seasonal pattern identification.
- **ML based Clustering and Insights:** Insights from specific, tailored ML models.
- **Predictive Analytics:** Case outcome prediction, risk assessment models, trend forecasting.
- **Network Analysis:** Perpetrator network detection, platform usage networks, agency collaboration patterns.

7 Conclusion

CaseLinker addresses a critical, underexplored gap in the technological ecosystem for protecting children from online exploitation: the need for interpretable, lightweight, case-level analysis tools that prioritize analyst well-being. Through deterministic feature extraction, two-stage clustering, and psychological safety-oriented design, the system transforms fragmented case archives into actionable intelligence—enabling pattern detection, priority triage, and strategic analysis that is impossible through manual review.

The system's open-source release and minimal infrastructure requirements democratize access to advanced case analysis, enabling resource-constrained agencies and non-profit advocacy organizations to benefit from capabilities previously available only through expensive enterprise platforms. By emphasizing interpretability over algorithmic sophistication, CaseLinker ensures that analytical outputs are auditable, explainable, and admissible—critical for legal proceedings.

Most importantly, CaseLinker recognizes that protecting children requires supporting those who do the protecting. Technology should not merely process cases efficiently; it should enable and support sustainable, humane investigation.

Availability: CaseLinker is open-source software available at https://github.com/mrinaalr/CaseLinker. A live demonstration is accessible at https://web-production-13a2.up.railway.app.

References

[1] National Center for Missing & Exploited Children. (2021). *2021 Reports by Electronic Service Providers*. NCMEC CyberTipline.

[2] Perez, L. M., Jones, J., Englert, D. R., & Sachau, D. (2010). Secondary traumatic stress and burnout among law enforcement investigators exposed to disturbing media images. *Journal of Police and Criminal Psychology*, 25(2), 113–124.

[3] Burns, C. M., Morley, J., Bradshaw, R., & Domene, J. (2008). The emotional impact on and coping strategies employed by police teams investigating internet child exploitation. *Traumatology*, 14(2), 20–31.

[4] Microsoft. (2009). *PhotoDNA*. Microsoft Corporation.

[5] Bursztein, E., Bright, T., DeLaune, M., Eliff, D. M., Hsu, N., Olson, L., Shehan, J., Thakur, M., & Thomas, K. (2019). Rethinking the Detection of Child Sexual Abuse Imagery on the Internet. In *Proceedings of the 2019 World Wide Web Conference (WWW '19)* (pp. 1–7). ACM. https://doi.org/10.1145/3308558.3313482

[6] Thorn. (2023). *Defend Children from Sexual Abuse*. Retrieved from https://www.thorn.org

[7] Meta. (2023). *Hasher-Matcher-Actioner (HMA)*. GitHub repository. https://github.com/facebook/ThreatExchange/tree/main/hasher-matcher-actioner

[8] Nuix. (2023). *Nuix Neo: AI-Powered Investigation Platform*. Nuix Limited.

[9] Magnet Forensics. (2023). *Magnet Axiom: Digital Investigation Platform*. Magnet Forensics Inc.

[10] Cellebrite. (2023). *UFED: Universal Forensic Extraction Device*. Cellebrite DI Ltd.

[11] BlackRainbow. (2023). *NIMBUS: Case Management System*. BlackRainbow Solutions.

[12] pdfplumber. (2023). PDF text extraction library. https://github.com/jsvine/pdfplumber.

[13] Bostock, M., Ogievetsky, V., & Heer, J. (2011). D3.js: Data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12), 2301–2309.

[14] Shneiderman, B. (1996). The eyes have it: A task by data type taxonomy for information visualizations. *Proceedings of IEEE Visualization*, 336–343.

[15] Kontostathis, A., Edwards, L., & Leatherman, A. (2010). Text mining and cybercrime. In *Text Mining: Applications and Theory* (pp. 1–20). John Wiley & Sons.

[16] Al-Nabki, M. W., Fidalgo, E., Alegre, E., & Alaiz-Rodriguez, R. (2023). Short text classification approach to identify child sexual exploitation material. *Scientific Reports*, 13, Article 16108. https://doi.org/10.1038/s41598-023-42902-8

[17] Ricanek, K., & Mahalingam, G. (2014). Age estimation from face images: Human vs. machine performance. In *2014 International Conference on Biometrics (ICB)* (pp. 1–8). IEEE.

[18] Laranjeira, C. M., Almeida, J., & Valle, E. (2022). Analysis Pipeline for Child Sexual Abuse Datasets. In *2022 ACM Conference on Fairness, Accountability, and Transparency (FAccT '22)* (pp. 2147–2158). ACM. https://doi.org/10.1145/3531146.3534636