LLM-Driven Knowledge Graph Construction for Human Smuggling Networks

Dipak Meher
George Mason University
Fairfax, USA
dmeher@gmu.edu

Carlotta Domeniconi
George Mason University
Fairfax, USA
cdomenic@gmu.edu

Guadalupe Correa-Cabrera
George Mason University
Fairfax, USA
gcorreac@gmu.edu

ABSTRACT

Human smuggling networks are increasingly adaptive and difficult to analyze. Legal case documents offer valuable insights but are unstructured, lexically dense, and filled with ambiguous or shifting references, posing challenges for automated knowledge graph (KG) construction. Existing KG methods often rely on static templates and lack coreference resolution, while recent LLM-based approaches frequently produce noisy, fragmented graphs due to hallucinations, and duplicate nodes caused by a lack of guided extraction. We propose CORE-KG, a modular framework for building interpretable KGs from legal texts. It uses a two-step pipeline: (1) type-aware coreference resolution via sequential, structured LLM prompts, and (2) entity and relationship extraction using domain-guided instructions, built on an adapted GraphRAG framework. CORE-KG reduces node duplication by 33.28% and legal noise by 38.37% compared to a GraphRAG-based baseline, resulting in cleaner and more coherent graph structures. These improvements make CORE-KG a strong foundation for analyzing complex criminal networks.

CCS CONCEPTS

• **Information systems** → **Information extraction**; • **Computing methodologies** → **Knowledge representation and reasoning**.

KEYWORDS

Knowledge Graph Construction, Coreference Resolution, Human Smuggling Networks

ACM Reference Format:
Dipak Meher, Carlotta Domeniconi, and Guadalupe Correa-Cabrera. 2025. LLM-Driven Knowledge Graph Construction for Human Smuggling Networks. In *KDD '25 Workshop SKow-LLM: Workshop on Structured Knowledge for Large Language Models, August 03–07, 2025, Toronto, CA.* ACM, New York, NY, USA, 13 pages. https://doi.org/XXXXXXX.XXXXXXX

Note: All example names in this paper are anonymized or fictitious to protect privacy. Full proper names and company's names are replaced with initials followed by a period (e.g., "Y.").

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
KDD'25 Workshop SKnow-LLM, August 03–07, 2025, Toronto, CA
© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Human smuggling has evolved into a complex and organized operation involving dynamic networks of actors, routes, vehicles, and intermediaries [6]. In human smuggling networks, smugglers facilitate human mobility through illicit means, typically in exchange for a fee and within the context of restrictive immigration policies. These networks exploit legal loopholes, adapt rapidly to enforcement policies, and often intersect with transnational criminal organizations. Understanding the structure and behavior of such illicit systems is essential for crafting effective policy, enhancing security, and preventing exploitation. Yet, much of the actionable intelligence about smuggling networks remains buried in unstructured legal texts such as court documents and case transcripts.

Despite growing interest from legal and social science communities, computational approaches for analyzing these documents remain underdeveloped. Entity references in unstructured legal text are often inconsistent, appearing as aliases, abbreviations, or role-based titles (e.g., "Officer Lewis" vs. "Defendant Lewis"). This complicates coreference resolution, entity normalization, and downstream tasks like extraction and knowledge graph construction.

Prior work has demonstrated the utility of knowledge graphs in legal investigations. For example, Mazepa et al.[25] and Shi et al.[29] used rule-based and regular-expression methods to construct graphs from homicide and indictment cases. However, these systems rely on static templates and lack the flexibility to resolve aliasing or surface-level entity variation, making them less robust in handling complex, multi-entity narratives.

While LLMs have shown strong potential in general knowledge extraction [20], their application to constructing knowledge graphs from human smuggling case narratives remains underexplored. In particular, challenges such as resolving surface-level redundancy across semantically equivalent mentions within typed entities—especially in heterogeneous, multi-entity documents like smuggling cases—are still largely unaddressed [16]. This represents a critical gap, as failure to consolidate such mentions leads to fragmented and redundant graph representations. Moreover, standard LLM-based extractors frequently misclassify or hallucinate entities and relations [15, 36], introducing semantic noise that impedes downstream analysis.

To address these challenges, we introduce CORE-KG —a modular, prompt-driven LLM-based framework for constructing precise and interpretable knowledge graphs from legal case documents. CORE-KG comprises two key components: (1) a *type-aware coreference resolution* module that consolidates semantically and contextually equivalent mentions within each entity type, and (2) a *knowledge graph construction* module that uses structured prompts to extract entities and relationships. These prompts include domain-specific

filtering instructions to suppress legal boilerplate, sequential type-wise extraction to minimize attention diffusion, and explicit type definitions to reduce classification ambiguity.

We evaluate CORE-KG on U.S. federal and state court cases related to human smuggling, drawn from publicly available legal proceedings. In the absence of annotated ground-truth graphs, we benchmark against a GraphRAG-based baseline [11], where both systems construct knowledge graphs from unstructured legal text using large language models applied over chunked inputs. On average, CORE-KG reduces node duplication by 33.28%, and legal noise by 38.37% in the resulting knowledge graphs. In a representative case study, CORE-KG identifies key actors, critical transit routes, transport vehicles, involved organizations, smuggled items, and communication methods, revealing structural insights essential for downstream analysis.[1]

To summarize, our key contributions are:

- We present CORE-KG, a modular, LLM-based framework that integrates type-aware coreference resolution and structured, domain-specific prompting to construct knowledge graphs from legal case documents on human smuggling.
- We introduce a prompt-based coreference resolution module that resolves semantically and contextually equivalent mentions across diverse entity types, enhancing consistency in structured knowledge extraction from complex narratives.
- CORE-KG outperforms a GraphRAG baseline, achieving a 33.28% reduction in node duplication, and 38.37% reduction in legal noise.

2 RELATED WORK

Knowledge graphs are powerful tools for transforming unstructured text into structured representations that facilitate reasoning, retrieval, and analysis [11, 37]. Recently, they have been applied to various tasks across domains such as education [7], life sciences [4], and construction safety management [12]. However, if not constructed carefully, they often suffer from issues such as duplicate nodes and fragmented structures, which significantly reduce their utility [15].

2.1 Knowledge Graph Construction

Traditionally, knowledge graphs have been built by first identifying important entities and their relationships from text, and then connecting them step by step. Entity extraction techniques include rule-based and dictionary-based methods [30], statistical and machine learning-based models [30], and open-domain or domain-specific extraction approaches [21, 24, 30]. For relation extraction, researchers have employed syntactic, lexical, and semantic modeling [17], ontology-based frameworks such as HowNet [22], and semi-supervised learning techniques [5]. While useful in specific domains, these methods often depend on hand-crafted extraction patterns and aim to preserve alignment between the graph and the source text.

More recently, LLM-based frameworks have emerged as powerful tools for constructing knowledge graphs in a prompt-driven manner [32, 36]. Kommineni et al. [18] proposed a semi-automated pipeline that integrates competency question generation, ontology design, and RAG-based triple extraction from scholarly texts. Zhang et al. [36] introduced a modular framework that performs open information extraction, schema definition, and canonicalization to reduce redundancy and improve consistency. While these approaches minimize manual effort, they often assume clean, unambiguous input and do not account for challenges like reference ambiguity or entity aliasing. This becomes particularly problematic in domains such as law, where entities frequently shift between aliases, roles, and pronouns.

To mitigate such challenges, coreference resolution plays a key role in maintaining graph coherence. Wang et al. [34], for instance, showed that even infrequent pronoun mentions can significantly affect the completeness of educational-domain knowledge graphs. This highlights the importance of integrating coreference mechanisms directly into the graph construction pipeline, especially when entity references are inconsistent or implicit.

Several efforts have also explored knowledge graph construction in criminal domains. Mazepa et al. [25] constructed a knowledge graph for homicide investigations using rule-based NLP pipelines and CoreNLP components. Shi et al. [29] developed a Neo4j-based graph for job-related crime indictments using regular-expression-based entity and relationship extraction. While both systems demonstrate the utility of knowledge graphs in legal and investigative contexts, they rely heavily on static templates and lack both coreference handling and modular prompting strategies, which limits their robustness in complex, ambiguous narratives.

Our work addresses this gap by introducing a prompt-driven, entity-type-specific coreference resolution strategy tailored for the modeling of criminal networks. This design improves semantic precision and node coherence in the resulting knowledge graphs, particularly in cases involving complex inter-entity relationships such as criminal networks, migration routes, and procedural actors.

2.2 Coreference Resolution

Several methods have been proposed to address the problem of duplicate nodes and fragmented structures through coreference resolution [19, 23]. Pogorilyy et al. [28] and Wu et al. [35] each proposed coreference resolution methods based on convolutional neural networks (CNNs), demonstrating improvements in modeling semantic and syntactic patterns in text.

Recent advances in large language models (LLMs) have brought new momentum to coreference resolution, particularly in low-resource settings where annotated data is scarce. Prompt-based approaches have shown that even with minimal supervision, LLMs can perform competitively by leveraging few-shot, zero-shot, or chain-of-thought strategies [9, 13]. These methods typically operate on datasets involving either a single entity type—often restricted to human mentions—or on general narrative text where entity-type distinctions are not explicitly modeled [9, 31]. While such setups enable LLMs to maintain focused attention and achieve high performance, they do not reflect the complexity of domain-specific texts where entities span diverse types and are referenced inconsistently across roles. As a result, the ability of LLMs to generalize across entity types and resolve fine-grained, role-shifting references remains largely untested.

[1] Our code is available at https://github.com/dipakmeher/CoreKG-HumanSmuggling.

Some progress has also been made in coreference resolution for legal texts. The work by Jia et al. [16] introduces a neural network-based model using ELMo, BiLSTM, and GCNs to resolve speaker-based coreference in court records. To the best of our knowledge, there is no prior research that investigates coreference resolution in the legal domain using large language models. Recent work has shown that instruction-tuned LLMs can achieve competitive performance in coreference resolution [20], but these evaluations are limited to standard narrative data and do not address the challenges posed by multi-entity legal documents.

3 METHOD

3.1 Method Overview

Our proposed pipeline is designed for modular knowledge graph construction (KG) from narrative-rich documents. While applicable across domains, we demonstrate its effectiveness on legal case files, with a focus on identifying actors, events, and entities involved in human smuggling networks. Specifically, we extract seven types of entities: *Person*, *Location*, *Organization*, *Route*, *Means of Transportation*, *Means of Communication*, and *Smuggled Items*. The pipeline consists of the following three major components:

- **Coreference Resolution:** We apply entity-type-aware coreference resolution to unify both surface and contextually grounded references to the same real-world entity (e.g., "Aaron Young," "Young," "Defendant"). This improves graph coherence and reduces redundancy, while preserving distinctions when context indicates separate referents.
- **Prompt Optimization for Entity and Relation Extraction:** We design a contextualized prompt that combines established prompting principles with domain-specific strategies for accurate entity and relation extraction. Although the framework is broadly applicable, we demonstrate its effectiveness for the task of building a KG for human smuggling networks. The prompt integrates sequential entity-type extraction, in-prompt type definitions, and explicit filtering instructions tailored to the downstream task of criminal network analysis. These components are organized in a chain-of-thought structure to reduce attention drift, improve type disambiguation, and eliminate irrelevant or noisy extractions.
- **KG Construction:** Entities and relationships extracted by a large language model (LLM), guided by a tailored prompt, are assembled into a structured knowledge graph using the GraphRAG knowledge graph construction module.

Since legal documents often contain procedural or statutory sections that are not central to the case narrative, we extract the *Opinion section* as input to our pipeline. This section typically contains the factual context and actor interactions most relevant to human smuggling cases. We apply coreference resolution to unify entity mentions, and then process the resolved text using the KG construction module to extract structured entities and relationships and build the final graph.

3.2 Coreference Resolution

Legal documents frequently refer to the same real-world entity using different surface forms throughout the text. For instance, a defendant introduced as "Aaron Young" may later be referred to as "Young," "the defendant," or functionally as "the driver." While this example concerns a `Person` entity type, similar patterns occur across other types as well. If these coreferent mentions are not resolved prior to the construction of the KG, they result in multiple disconnected nodes representing the same entity, fragmenting relationships and undermining graph interpretability. Such fragmentation dilutes the structural coherence and weakens the utility of the graph for downstream analysis.

3.2.1 Coreference Resolution Module. To address entity fragmentation, our Coreference Resolution module leverages the contextual reasoning capabilities of large language models (LLMs) [14, 33] to unify different surface forms and contextually grounded references to the same real-world entity into a single, consistent representation. For example, references like "Young," and "the defendant," are resolved to one canonical form. This unification ensures that all relationships tied to that entity are aggregated under a single node in the KG, improving both structural consistency (avoiding redundant nodes) and analytical clarity (making entity relationships easier to interpret).

The module operates over the Opinion section of each legal case and performs coreference resolution separately for each entity type with the help of an LLM. LLMs, trained on large and diverse text corpora, excel at understanding context, reasoning over long spans of text, and linking semantically related phrases [3, 10, 26]. However, they perform best when their attention is guided toward a specific objective. When multiple types are resolved simultaneously, the model's attention becomes diffusely distributed across semantically unrelated categories (e.g., persons, route, organizations, etc.) [1]. Since LLMs rely on self-attention mechanisms, which are designed to dynamically assign importance to different spans of text, this broad distribution leads to diluted focus, making it harder for the model to form accurate coreference chains, i.e., correctly linking all references to the same entity.

From a representation learning perspective, mixing multiple semantic types in a single pass increases the risk of type drift (where an entity is assigned the wrong category) and feature entanglement (where features from different types interfere with one another) [38]. These errors can propagate into downstream tasks like entity linking or graph construction, leading to noisy or incomplete KGs. This issue is exacerbated when surface forms are ambiguous or overlap semantically. For example, the phrase "The Camp" might refer to a temporary location used for migrant holding (*a Location*) but could also be misclassified as an *Organization*, especially when entity boundaries are vague or context is minimal. Similarly, "the van" might refer to a vehicle used for transport (*Means of Transportation*), but in some contexts, it may be interpreted as the location where migrants are held (*Location*). Such ambiguity in surface forms often results in incorrect type assignments and fragmented entity clusters in the graph.

To mitigate these risks, we adopt a type-wise sequential resolution strategy. The model resolves one entity type at a time, which

reduces cross-type interference and allows more accurate consolidation of coreferences. The pipeline operates as follows:

(1) The filtered opinion text is first passed to the LLM along with a prompt tailored to resolve `Person` entities. Mentions like "Young," "the defendant," and "the driver" are linked and replaced with their canonical reference.
(2) The output from this step is then used in the next stage, where a `Location` prompt resolves place-based mentions like "Laredo" and "Laredo Texas."
(3) The process continues in sequence for the remaining entity types: `Routes`, `Organization`, `Means of Transportation`, `Means of Communication`, and `Smuggled Items`.

This targeted approach improves semantic precision by ensuring correct type assignment and enhances structural fidelity through clear, non-overlapping entity representation in the resulting graph. With input from a subject matter expert, we limit the resolution process to seven entity types, helping the model avoid overgeneralization and stay aligned with the goals of criminal network modeling.

3.2.2 Coreference Resolution Prompt Design. Since coreference resolution is performed separately for each entity type, we design a dedicated prompt for every entity type of interest. Each prompt is paired with the input text to guide the language model in resolving coreferences for that specific category only.

Extensive prompt optimization is performed independently for each entity type to maximize the quality and completeness of coreference resolution. The prompts are designed through iterative refinement and evaluation, ensuring that they generalize well across diverse legal cases. Particular attention is given to capturing the wide variability in surface forms within each category. For example, `Person` entities may appear as full names, shortened names, aliases, and roles (e.g., `Defendant`, `Driver`). `Location` entities may be expressed with varying granularity (e.g., `Laredo`, `"Laredo, Texas"`, while `Routes` can be denoted through formal identifiers or colloquial expressions (e.g., `Interstate 35`, `I-35`). Similarly, `Organizations` may include government bodies, company names, or abbreviations, and `Means of Transportation` often blend object and ownership references (e.g., `Trailer`, `Tractor-Trailer`). In designing these prompts, we intentionally avoided overfitting to any fixed phrase list and instead focused on creating entity-type-specific guidelines and few-shot examples that reflect the semantic diversity and ambiguity present in real legal documents. As a result, the prompts are intended to support generalization to unseen or unexpected surface forms. In Section 5.1 we discuss several qualitative examples where the model correctly resolved such references, including collective terms like "relatives of the aliens".

To maintain consistency and precision across these varied patterns, each prompt follows a structured format containing the below key components:

- **Persona definition**: The LLM is assigned the role of a highly accurate and rule-following coreference resolution system.
- **Clear task description**: The prompt clearly states that the LLM must resolve all coreferences without summarizing, rephrasing, or altering the input text in any way.
- **Contextual information**: The prompt explains the downstream use of the resolved text, such as for building knowledge graphs to analyze human smuggling networks.
- **Entity-type-specific resolution rules**: Tailored resolution guidelines are provided for each entity type based on their linguistic patterns. For instance, the rules for the `Person` entity type address variations such as full names, abbreviations, last names, aliases, and role-based references (e.g., "the driver", "the defendant").
- **Few-shot examples**: Illustrative examples are provided to show the model how to correctly resolve coreferences in different realistic scenarios.

An example of the coreference resolution prompt used for the `Person` entity type is provided in Appendix Figure 6. This structured and detailed prompt design ensures that the model receives precise guidance, thereby improving the accuracy and consistency of coreference resolution across diverse legal documents.

3.3 Entity-Relationship Extraction and KG Construction

The coreference-resolved Opinion section serves as input to the Knowledge Graph Construction (KGC) module of the GraphRAG framework [11]. GraphRAG is a modular retrieval-augmented generation (RAG) system that combines structured knowledge graph construction with neural retrieval and language model-based generation. It has gained traction for its ability to mitigate hallucination and improve factual grounding in downstream tasks by anchoring generation to a graph-based representation of the input text.

GraphRAG consists of two core components: (1) a KGC module that extracts entities and relationships to form a structured graph, and (2) a retrieval-augmented generation module that leverages this graph as an index for response generation. This work concentrates specifically on the KGC component, which we extend and tailor to the processing of legal case files for human smuggling network analysis.

Each input document is divided into overlapping chunks of 300 tokens, which are passed to the LLM along with an extraction prompt optimized for the legal domain of human smuggling cases. The model outputs entity–relationship triples, which are aggregated across chunks. GraphRAG performs basic post-processing—merging entities based on exact string and type matches—and assembles the graph using the `NetworkX` library. Final outputs are serialized in both GraphML and Parquet formats to support further analysis and visualization via tools like Gephi or the GraphRAG visualizer.

3.3.1 Prompt Tuning for Entity and Relationship Extraction. The prompt for entity and relationship extraction follows the same principled design strategies outlined in Section 3.2.2. However, unlike the coreference resolution module—where prompts are applied sequentially for each entity type—GraphRAG's KGC component requires a single unified prompt to extract all relevant entity types and their relationships in one pass.

To accommodate this, the prompt is carefully designed to balance breadth and precision across the seven targeted types, enabling the LLM to generate structured triples with minimal noise and high task

relevance. It incorporates the same components as the coreference prompts—namely, persona definition, task description, contextual framing, extraction steps, output format specification, and few-shot examples—as detailed in Section 3.2.2. An example of the final prompt used for entity and relationship extraction is shown in Appendix Figure 3.

To further improve the quality of entity and relationship extraction for human smuggling network analysis, we introduce several additional modifications to the prompt design, as described below.

Sequential Entity Extraction to Reduce Attention Distribution. GraphRAG's default prompt extracts all entity types jointly, which can lead to attention spread across multiple categories—causing the model to miss entities, assign incorrect types, or overlook fine-grained distinctions [2]. These issues are especially pronounced in legal texts, where diverse entity types frequently co-occur within complex, narrative structures.

To mitigate this, we introduced a strict ordering mechanism within the prompt: the model is explicitly instructed to first extract entities of type *Person*, followed by *Location*, *Routes*, *Organization*, and so on. Only after completing entity extraction in the specified sequence does the model proceed to identify relationships among the extracted entities. By enforcing sequential extraction by entity type, we significantly reduce competition for attention within the model, enabling it to capture all mentions of one entity category before shifting focus to the next—thereby improving both the precision of extracted entities and the completeness of relationships, which directly contributes to cleaner and more reliable knowledge graph construction.

Filtering High-Frequency Irrelevant Entities. Another significant challenge arises from the presence of high-frequency but contextually irrelevant entities within legal documents, particularly government-related entities such as courts, juries, appeals, and judicial offices. These entities are typically classified under the *Organization* type during extraction. However, if included in the final knowledge graph, these non-critical entities tend to dominate the node distribution and inflate relationship statistics due to their frequent mentions in legal proceedings, which introduces noise and obscures the true patterns of interaction among the smuggling participants.

To address this issue during extraction itself, we introduced an explicit filtering instruction within the prompt. After completing the entity and relationship extraction steps, the LLM is guided to identify all government-related nodes based on predefined criteria and remove them before producing the final output. By integrating this filtering at the extraction stage, we eliminate the need for separate post-processing to clean the graph. This early-stage filtering improves the overall quality, clarity, and analytical usefulness of the resulting knowledge graphs by ensuring that only domain-relevant entities are retained.

Entity Type Definitions to Mitigate Overgeneralization Bias. LLMs, despite being trained on large and diverse datasets, often exhibit overgeneralization bias, where entities are misclassified based on frequent training patterns rather than the actual context [8, 27]. For instance, if certain words often appear near geographic names during pretraining, the model may wrongly label unrelated terms like event names or organizational units as *Location* entities.

This issue is amplified in legal texts, where specialized terms and subtle context influence type classification. To address this, we embed precise definitions of all seven entity types directly in the prompt, clearly distinguishing *Person*, *Location*, *Route*, and *Organization* entities, along with examples. These definitions help the model make context-aware decisions, improving extraction accuracy and reducing type confusion in the resulting knowledge graph.

4 EXPERIMENTAL SETUP

We aim to answer the following research questions:

- **RQ1:** Does incorporating prompt-based coreference resolution reduce node duplication in knowledge graphs constructed from human smuggling case documents?
- **RQ2:** Does CORE-KG—our fully adapted pipeline incorporating coreference resolution, prompt tuning, and legal-specific filtering—generate more relevant and concise knowledge graphs than the GraphRAG-based baseline?

4.1 Dataset

To evaluate our system we use judicial cases accessed via the academic search engine Nexis Uni, which is available to us through the library of our Institution. We focus on cases related to human smuggling networks. These documents are drawn from federal and state court proceedings filed between 1994 and 2024, providing a diverse and representative sample of criminal smuggling activities across jurisdictions. All information referenced in these cases is part of the public domain.

However, legal case files often include extensive procedural, statutory, and administrative content, which introduces noise and redundancy when constructing knowledge graphs that aim to capture the structure of criminal networks. To mitigate this, we apply targeted preprocessing to extract only the section entitled "Opinion" from each case. This section contains the main factual narrative, describing the people involved, the routes used, the items transported, and the sequence of events. These elements are directly relevant for building meaningful and interpretable knowledge graphs. All methods compared in our experiments are provided with the same input text from the Opinion section, hereafter referred to as the input document.

In our preliminary experiments we have randomly selected a sample of 20 cases (retrieved from the Nexis Uni database using the query: "*human smuggling OR alien smuggling*"), in which the Opinion section contains approximately 2000 words. Although a larger sample size is needed for comprehensive analysis, the selected cases sufficiently reflect the narrative complexity and are suitable for a preliminary assessment of the system's capacity to extract key entities and identify meaningful relationships and interactions.

4.2 Implementation

We employ the LLaMA 3.3 70B model for both coreference resolution and knowledge graph construction. To ensure reproducibility and minimize stochastic variation during inference, the temperature parameter is set to zero. The model is served locally using

the Ollama framework, allowing us to operate entirely within an open-source environment without relying on commercial or closed-source APIs. All experiments were conducted on an NVIDIA A100 GPU with 80GB of memory.

Our codebase is implemented in Python 3.12. For GraphRAG [11], we follow the setup instructions provided in the official GitHub repository and use version 0.3.2. We configured GraphRAG to use 300-token chunks and specified the nomic-embed-text embedding model; however, embeddings were not utilized, as the knowledge graph construction process does not rely on them.

4.3 Baselines and Experimental Design

Given the lack of prior work on structured knowledge graph construction from legal texts—especially for criminal network analysis such as human smuggling—we evaluate our system against a baseline derived from the GraphRAG framework [11], which is widely adopted for knowledge-intensive tasks. We compare two system variants:

- **GraphRAG (Baseline):** A minimally adapted version of the default GraphRAG prompt [11]. It includes the addition of the seven targeted entity types and a few-shot example to guide extraction, but retains the original model behavior, output format, and processing flow.
- **CORE-KG (Ours):** A fully adapted pipeline that integrates several domain-specific enhancements. These include (1) prompt-based coreference resolution, (2) sequential entity extraction to reduce attention dilution, (3) in-prompt entity type definitions and negative examples to guide classification, and (4) legal-specific filtering instructions to suppress noisy or irrelevant entities.

Since no annotated ground truth exists for structured knowledge graphs in this domain, we adopt both a qualitative and quantitative evaluation framework. Both systems are applied to the same 20 input documents. The resulting graphs are evaluated as follows:

- **Node Duplication:** We compute the number of redundant nodes representing the same real-world entity (e.g., `Young`, `Aaron Young`, `Defendant`). This process is partially automated using intra-type fuzzy matching (see Section 5.1).
- **Noise Reduction:** We document the inclusion of generic legal boilerplate or irrelevant procedural terms (e.g., `Judicial Proceedings`, `Appeal Process`, `Court`) that do not aid downstream smuggling network analysis, and assess their suppression across systems.
- **Type Assignment Reliability:** We qualitatively examine common misclassifications of entities (e.g., labeling "The Safehouse" as an organization instead of a location) and evaluate the impact of prompt-level refinements in reducing such errors.

This evaluation protocol enables a focused comparison between the baseline and CORE-KG outputs, isolating the effects of coreference resolution and prompt engineering in building domain-aligned, analytically useful knowledge graphs.

5 RESULTS

5.1 RQ1: Impact of Coreference Resolution on Node

To evaluate the impact of coreference resolution, we analyze the extent of node duplication in both the baseline system and our CORE-KG pipeline. Duplicate nodes are defined as semantically equivalent entities that appear multiple times in the graph under different surface forms (e.g., `Young`, `Aaron Young`).

Duplicate Node Detection. Duplicate node detection is performed through a two-stage process. First, we apply fuzzy string matching to the extracted entities using the `partial_ratio` function from the `RapidFuzz` library, retaining all intra-type entity pairs with similarity scores at or above a 75% threshold. These pairs are used to construct an undirected similarity graph, from which clusters are extracted as connected components—each assumed to represent a single real-world entity. For example, mentions such as `white pickup truck`, `stolen white pickup truck`, and `white older Ford pickup truck` would be grouped into the same cluster. In the second stage, a manual review is conducted by a subject matter expert to correct false positives—i.e., cases where distinct entities were incorrectly grouped together.

The final node duplication count is computed as $\sum_{C_i}(|C_i| - 1)$, where C_i denotes a cluster of mentions referring to the same entity. To allow for comparability across graphs of varying sizes, we report the *node duplication rate* as the number of redundant nodes divided by the total number of nodes in the graph. This value represents the proportion of all entity nodes that are considered duplicates. *Absolute Drop* is computed as the direct difference between the baseline and CORE-KG scores. *Relative Improvement* is calculated as the percentage reduction relative to the baseline: (Baseline − CORE-KG)/Baseline × 100.

Noise Detection. To enhance the quality and interpretability of the generated knowledge graphs, we conduct a systematic manual validation of the extracted entities in each case. This validation, performed by a domain expert, involves identifying legal boilerplate or procedural terms (e.g., `Court`, `Appeal`, `Judicial Proceedings`) that do not contribute to the structural understanding of the smuggling network. Although frequent in legal texts, such entities are deemed semantically irrelevant for downstream analysis. We quantify noise using the *noise rate*, defined as the number of non-informative nodes divided by the total number of nodes in the graph, multiplied by 100.

Metric	Baseline (%)	CORE-KG (%)	Absolute Drop (%)	Relative Improvement (%)
Node Duplication Rate	30.38	20.27	10.11	33.28
Noise Rate	27.41	16.89	10.52	38.37

Table 1: Comparison of CORE-KG and the baseline across three key evaluation metrics aggregated over 20 legal cases.

Table 1 presents the average duplication statistics across 20 legal case graphs, comparing the graphs generated by the baseline GraphRAG system with those produced by our CORE-KG pipeline. We observe a substantial reduction in node duplication, driven by the improvements introduced through entity-type-aware prompt-based coreference resolution. In the baseline, 30.38% of all nodes in the graph were redundant mentions of entities already captured

by another node in their cluster. CORE-KG reduces this to 20.27%, yielding a 33.28% relative improvement. These results underscore the effectiveness of our pipeline in resolving coreference, collapsing redundant structures, and producing cleaner, more coherent knowledge graphs. Detailed case-wise comparisons for all three metrics across the 20 legal cases are presented in Appendix A, showing consistent performance improvements of CORE-KG over the baseline.

5.1.1 CORE-KG's Context-Aware and Generalization Capability in Coreference Resolution. While conducting error analysis on unresolved duplicates, we found that not all apparent misses were true failures. For example, in Case 6, phrases like "undocumented alien," "illegal trafficking in aliens," and "smuggling aliens" were not merged by CORE-KG, despite their surface similarity. Closer inspection revealed that these referred to distinct concepts: a specific individual in custody, a general criminal activity, and broader trafficking patterns along a route. By preserving these distinctions, CORE-KG demonstrated sensitivity to discourse context and avoided spurious generalization.

In Case 13, mentions such as "relatives," "relatives of the aliens," and "family members"—all referring to individuals in New York connected to Daniel Arturo Reyes Navarro—were initially extracted as separate nodes by the baseline. CORE-KG correctly unified them into a single node labeled Daniel Arturo Reyes Navarro's Relatives, capturing indirect, collective references absent from the prompt.

These examples provide strong qualitative evidence that CORE-KG does not indiscriminately cluster surface-similar mentions. Instead, it applies contextual judgment—guided by structured prompts and LLM inference—to balance generalization with precision across noisy, heterogeneous legal narratives.

5.1.2 Error Analysis: Missed Coreference Resolutions. That said, we did observe a few genuine misses. For example, in Case 15, the mentions "United States–Mexican border" and "border" were not clustered, and in Case 10, "United States government" and "United States" were treated as separate entities. We acknowledge these as true missed coreference cases. A likely reason is that legal documents often contain high lexical density and institutional specificity, where certain entity mentions—although contextually coreferent—appear in subtly distinct syntactic and semantic frames. In such contexts, the LLM may treat "border" as a general geographic location and "United States–Mexican border" as a formal, compound entity. Similarly, "United States" may refer either to the country as a location or the government as an actor, depending on local context. Without stronger global discourse tracking or reinforcement of named entity role alignment, these cases may fall outside the model's resolution threshold.

These observations suggest opportunities for improving prompt sensitivity to geopolitical and institutionally anchored entities, particularly in domains with formal legal discourse.

5.2 RQ2: Does CORE-KG produce more relevant and concise knowledge graphs compared to the baseline?

In this section, we qualitatively assess the resulting knowledge graphs by evaluating the impact of prompt design across three dimensions: structural gain, reduction of noisy legal entity extractions, and resolution of entity type misclassifications.

5.2.1 Structural Gains from Prompt-Guided Graph Construction. To assess the structural benefits of CORE-KG, we compare its output against the baseline GraphRAG system using a representative case 2. Appendix Figures 4 and 5 show the resulting knowledge graphs from both systems. In these visualizations, relationship labels are omitted for clarity, and node size reflects the number of connected relationships, serving as a proxy for centrality.

In the baseline graph, the primary actor—`A.Y.`—is fragmented across two separate nodes: `Y.`, and `A.Y.`. These nodes are only indirectly linked, resulting in a disjointed structure that inflates graph complexity and weakens narrative continuity. Notably, the most central node in the baseline graph is `Y.`, despite the existence of multiple redundant mentions referring to the same individual. In contrast, CORE-KG successfully consolidates all references into a single, canonical `A.Y.` node, producing a cleaner and semantically faithful representation of the actor's role.

Further structural improvements are evident in the connectivity of key locations. In the baseline graph, `Laredo, Texas`—a critical geographic location in the smuggling operation is disconnected, illustrating how legal noise and lack of guidance in extraction can obscure essential entities. In CORE-KG, however, `Laredo, Texas` is clearly connected to relevant entities, including `A.Y.`, `Interstate 35`, `Illegal Aliens`, and `Agent J.C.D.A.`, reflecting a more complete and meaningful network of interactions.

These improvements illustrate how CORE-KG's guided extraction strategy produces more coherent, focused, and narratively aligned graphs from complex legal texts.

5.2.2 Reducing Legal Noise Through Prompt-Guided Filtering. Since the input text is drawn from legal opinions, the baseline graph frequently includes noisy legal terms that are common in such documents but irrelevant to smuggling network analysis. In the qualitative example from Case 2, the baseline system extracted 28 such entities—including `state court`, `district court`, and `court of appeals`—which dilute the semantic clarity of the graph, as visualized in Figure 2 (Appendix). These nodes introduce noise and obscure the core smuggling narrative. Additional examples of such irrelevant legal entities are provided in Table 2.

A key limitation of the baseline system is that it prompts the LLM to extract all seven entity types simultaneously, which leads to attention dilution and type confusion. CORE-KG addresses this through two key improvements. First, it includes explicit filtering instructions that direct the model to ignore procedural and legal boilerplate terms. Second, it employs in-prompt sequential extraction, allowing the model to focus on one entity type at a time in a fixed order. This structured design significantly improves extraction accuracy. The prompt used for extraction is shown in Appendix Figure 3.

In the same legal case, CORE-KG extracted no legal noise entities, yielding a clean and focused graph. Quantitatively, CORE-KG reduces the overall legal noise rate from 27.41% to 16.89%, an absolute improvement of 10.52% points (see Table 1). This demonstrates the model's improved ability to focus on domain-relevant entities and avoid cluttering the graph with noisy nodes.

Issue Type	Observed in Baseline	CORE-KG Behavior and Reason for Improvement
Duplicate Entities	Multiple variants like `Y.`, `A.Y.`; `J.C.D.A`, `Agent D.A.`; `Laredo`, `Laredo Texas`; `I-35`, `Interstate 35`, `Route from Laredo to Dallas`; `B.`, `Y.B.`.	Consolidated into unified nodes through entity-type-specific coreference resolution: `A.Y.`, `J.C.D.A`, `Laredo Texas`, `Interstate 35`, `Y.B.`.
Generic Legal Entities	Entities like `court`, `district court`, `court of appeals`, `state court`, `appeal`, `appeal process`, `judgment of acquittal`, `motion for judgment of acquittal`, and `plain error standard` clutter the graph and lack task relevance.	Removed through in-prompt filtering and reasoning-based instructions designed to retain only entities relevant to downstream analysis of smuggling networks.
Entity Type Misclassification	Examples include `Court Case` and `Federal Proceedings` misclassified as `Location`; `United States Magistrate Judge` mislabeled as `Organization` instead of `Person`; `Jamco` tagged as `Location` rather than `Organization`; and abstract terms like `Verdict` or `Ineffective Assistance` incorrectly extracted as `Smuggled Items`.	CORE-KG reduces such misclassifications through sequential, type-specific extraction and explicit in-prompt definitions that help the LLM disambiguate and align entity roles correctly.

Table 2: Qualitative Comparison of Prompt Effectiveness: CORE-KG vs. Baseline on *a representative legal case*

5.2.3 Impact of Prompt Design on Type Misclassification. Another critical improvement observed in CORE-KG pertains to entity type classification. In the baseline system, the language model often misclassifies entities due to vague type boundaries and insufficient contextual guidance. For example, `United States Magistrate Judge` is incorrectly labeled as an `Organization` instead of a `Person`, and `South Laredo` is tagged as a `Route` rather than a `Location`. In the representative case, the model further misclassifies legal terms such as `EVIDENCE` and `Verdict` as `Smuggled Items`, and `APPEAL PROCESS` as a `Route`. Additionally, entities like `J.` are incorrectly extracted as `Location` instead of their correct type, `Organization`. These errors are compounded by the baseline's tendency to assign types to legal boilerplate terms that should not have been extracted in the first place. Such misclassifications distort the semantic structure of the graph and hinder downstream analysis of actor–resource–event dynamics central to understanding smuggling networks.

CORE-KG substantially reduces these errors through two key prompt-level interventions. First, it employs sequential type-wise extraction, instructing the model to focus on one entity type at a time. This design minimizes cross-type confusion, especially in cases with ambiguous lexical cues. Second, the prompt provides explicit, context-sensitive definitions for each entity type, clearly distinguishing, for example, between actionable smuggled items and procedural legal terminology. These refinements constrain the model's output space and promote more accurate and schema-aligned entity typing.

In the representative case, CORE-KG not only avoids extracting out-of-scope legal concepts but also correctly resolves `J.` to its full form, `J.I. Inc.`, and classifies it as an `Organization`—a classification consistent with its semantic role in the document. These improvements collectively enhance the accuracy, interpretability, and task relevance of the resulting knowledge graph.

6 CONCLUSION

In this work, we presented CORE-KG, a modular LLM-driven framework for constructing interpretable knowledge graphs from legal case documents describing human smuggling networks. By integrating type-aware prompt-based coreference resolution and domain-specific prompt design, CORE-KG addresses key challenges such as duplicate nodes, noisy legal entities, and entity misclassification—common issues in both rule-based and general-purpose LLM-based systems. Through evaluation on real-world smuggling cases, CORE-KG demonstrated significant improvements over a GraphRAG-based baseline, achieving a 33.28% reduction in node duplication, and 38.37% reduction in legal noise. These gains result in cleaner, more coherent graphs that better support downstream analysis of criminal networks. Our results underscore the importance of structured prompting, entity-type separation, and legal-domain adaptation for improving the quality of LLM-generated knowledge graphs. We believe CORE-KG lays a strong foundation for future work in automated analysis of human smuggling networks, including group discovery, entity role identification within groups, temporal graph evolution, and event prediction from legal texts and other complex narrative sources.

ACKNOWLEDGMENT

This material is based upon work supported by the U.S. Department of Homeland Security under Grant Award Number 17STCIN00001-08-00. Disclaimer: The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security.

REFERENCES

[1] Sahar Abdelnabi, Aideen Fay, Giovanni Cherubin, Ahmed Salem, Mario Fritz, and Andrew Paverd. 2024. Are you still on track!? Catching LLM Task Drift with Activations. *arXiv preprint arXiv:2406.00799* (2024).

[2] Sahar Abdelnabi, Aideen Fay, Giovanni Cherubin, Ahmed Salem, Mario Fritz, and Andrew Paverd. 2025. Get My Drift? Catching LLM Task Drift with Activation Deltas. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. IEEE, 43–67.

[3] Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. 2024. Make your llm fully utilize the context. *Advances in Neural Information Processing Systems* 37 (2024), 62160–62188.

[4] Tiffany J Callahan, Ignacio J Tripodi, Adrianne L Stefanski, Luca Cappelletti, Sanya B Taneja, Jordan M Wyrwa, Elena Casiraghi, Nicolas A Matentzoglu, Justin Reese, Jonathan C Silverstein, et al. 2024. An open source knowledge graph ecosystem for the life sciences. *Scientific Data* 11, 1 (2024), 363.

[5] Andrew Carlson, Justin Betteridge, Richard C Wang, Estevam R Hruschka Jr, and Tom M Mitchell. 2010. Coupled semi-supervised learning for information extraction. In *Proceedings of the third ACM international conference on Web search and data mining*. 101–110.

[6] Sara Carrasco-Granger, Rut Bermejo-Casado, and Isabel Bazaga Fernández. 2025. Scapegoating Human Smugglers: How Migrants' Accounts Challenge the EU's Dominant Narrative. *Journal of Immigrant & Refugee Studies* (2025), 1–13.

[7] Penghe Chen, Yu Lu, Vincent W Zheng, Xiyang Chen, and Boda Yang. 2018. Knowedu: A system to construct knowledge graph for education. *Ieee Access* 6 (2018), 31553–31563.

[8] Sunhao Dai, Chen Xu, Shicheng Xu, Liang Pang, Zhenhua Dong, and Jun Xu. 2024. Bias and unfairness in information retrieval systems: New challenges in the llm era. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 6437–6447.

[9] Mridusmita Das and Apurbalal Senapati. 2024. Co-reference Resolution in Prompt Engineering. *Procedia Computer Science* 244 (2024), 194–201.

[10] Yiran Ding, Li Lyna Zhang, Chengruidong Zhang, Yuanyuan Xu, Ning Shang, Jiahang Xu, Fan Yang, and Mao Yang. 2024. Longrope: Extending llm context window beyond 2 million tokens. *arXiv preprint arXiv:2402.13753* (2024).

[11] Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. 2024. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130* (2024).

[12] Weili Fang, Ling Ma, Peter ED Love, Hanbin Luo, Lieyun Ding, and AO Zhou. 2020. Knowledge graph for identifying hazards on construction sites: Integrating computer vision with ontology. *Automation in Construction* 119 (2020), 103310.

[13] Yujian Gan, Juntao Yu, and Massimo Poesio. 2024. Assessing the capabilities of large language models in coreference: An evaluation. In *Joint 30th International Conference on Computational Linguistics and 14th International Conference on Language Resources and Evaluation, LREC-COLING 2024*. European Language Resources Association (ELRA), 1645–1665.

[14] Alex Havrilla, Sharath Raparthy, Christoforus Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskyi, Eric Hambro, and Roberta Raileanu. 2024. Glore: When, where, and how to improve llm reasoning via global and local refinements. *arXiv preprint arXiv:2402.10963* (2024).

[15] Elwin Huaman, Elias Kärle, and Dieter Fensel. 2020. Duplication detection in knowledge graphs: Literature and tools. *arXiv preprint arXiv:2004.08257* (2020).

[16] Donghong Ji, Jun Gao, Hao Fei, Chong Teng, and Yafeng Ren. 2020. A deep neural network model for speakers coreference resolution in legal texts. *Information Processing & Management* 57, 6 (2020), 102365.

[17] Nanda Kambhatla. 2004. Combining lexical, syntactic, and semantic features with maximum entropy models for information extraction. In *Proceedings of the ACL interactive poster and demonstration sessions*. 178–181.

[18] Vamsi Krishna Kommineni, Birgitta König-Ries, and Sheeba Samuel. 2024. From human experts to machines: An LLM supported approach to ontology and knowledge graph construction. *arXiv preprint arXiv:2403.08345* (2024).

[19] Kusum Lata, Pardeep Singh, and Kamlesh Dutta. 2022. Mention detection in coreference resolution: survey. *Applied Intelligence* 52, 9 (2022), 9816–9860.

[20] Nghia T Le and Alan Ritter. 2023. Are Large Language Models Robust Coreference Resolvers? *arXiv preprint arXiv:2305.14489* (2023).

[21] Xiao Ling and Daniel Weld. 2012. Fine-grained entity recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 26. 94–100.

[22] Kebin Liu, Fang Li, Lei Liu, and Ying Han. 2007. Implementation of a kernel-based Chinese relation extraction system. *Jisuanji Yanjiu yu Fazhan(Computer Research and Development)* 44, 8 (2007), 1406–1411.

[23] Ruicheng Liu, Rui Mao, Anh Tuan Luu, and Erik Cambria. 2023. A brief survey on recent advances in coreference resolution. *Artificial Intelligence Review* 56, 12 (2023), 14439–14481.

[24] Xiaohua Liu, Shaodian Zhang, Furu Wei, and Ming Zhou. 2011. Recognizing named entities in tweets. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*. 359–367.

[25] Svitlana Mazepa, Victoria Vysotska, Danylo Ivanchyshyn, Lyubomyr Chyrun, Vadim Schuchmann, and Yuriy Ryshkovets. 2022. Relationships Knowledge Graphs Construction Between Evidence Based on Crime Reports. In *2022 IEEE 17th International Conference on Computer Sciences and Information Technologies (CSIT)*. IEEE, 165–171.

[26] Dipak Falgun Meher, Ajay Krishna Vajjala, and David S Rosenblum. [n. d.]. Understanding User Behavior in Cross-Domain Recommendation: An LLM-Based Approach. ([n. d.]).

[27] Uwe Peters and Benjamin Chin-Yee. 2025. Generalization bias in large language model summarization of scientific research. *Royal Society Open Science* 12, 4 (2025), 241776.

[28] Sergiy Pogorilyy and Artem Kramov. 2019. Coreference resolution method using a convolutional neural network. In *2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT)*. IEEE, 397–401.

[29] Yong Shi, Wenlu An, Jiayu Xue, and Yi Qu. 2022. A knowledge graph constructed for job-related crimes. *Procedia Computer Science* 199 (2022), 540–547.

[30] Peng Sun, Xuezhen Yang, Xiaobing Zhao, and Zhijuan Wang. 2018. An overview of named entity recognition. In *2018 International Conference on Asian Language Processing (IALP)*. IEEE, 273–278.

[31] Hieu-Dai Tran, Duc-Vu Nguyen, and Ngan Luu-Thuy Nguyen. 2025. Coreference Resolution for Vietnamese Narrative Texts. *arXiv preprint arXiv:2504.19606* (2025).

[32] Julio Vizcarra, Shuichiro Haruta, and Mori Kurokawa. 2024. Representing the interaction between users and products via llm-assisted knowledge graph construction. In *2024 IEEE 18th International Conference on Semantic Computing (ICSC)*. IEEE, 231–232.

[33] Boshi Wang, Xiang Yue, and Huan Sun. 2023. Can ChatGPT defend its belief in truth? evaluating LLM reasoning via debate. *arXiv preprint arXiv:2305.13160* (2023).

[34] Tai Wang and Huan Li. 2020. Coreference resolution improves educational knowledge graph construction. In *2020 IEEE International Conference on Knowledge Graph (ICKG)*. IEEE, 629–634.

[35] Jheng-Long Wu and Wei-Yun Ma. 2017. A deep learning framework for coreference resolution based on convolutional neural network. In *2017 IEEE 11th International Conference on Semantic Computing (ICSC)*. IEEE, 61–64.

[36] Bowen Zhang and Harold Soh. 2024. Extract, define, canonicalize: An llm-based framework for knowledge graph construction. *arXiv preprint arXiv:2404.03868* (2024).

[37] Lingfeng Zhong, Jia Wu, Qian Li, Hao Peng, and Xindong Wu. 2023. A comprehensive survey on automatic knowledge graph construction. *Comput. Surveys* 56, 4 (2023), 1–62.

[38] Wenxuan Zhou, Sheng Zhang, Yu Gu, Muhao Chen, and Hoifung Poon. 2023. Universalner: Targeted distillation from large language models for open named entity recognition. *arXiv preprint arXiv:2308.03279* (2023).

A PER-CASE DUPLICATION AND NOISE METRICS

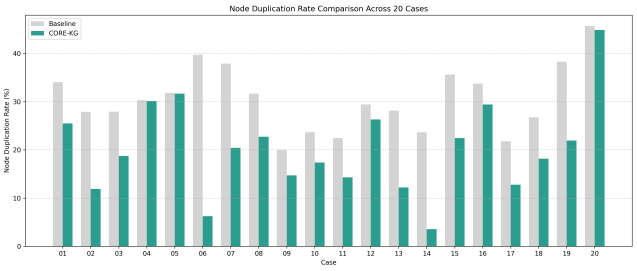

Figure 1: Node duplication rate comparison between baseline and CORE-KG across 20 legal cases.

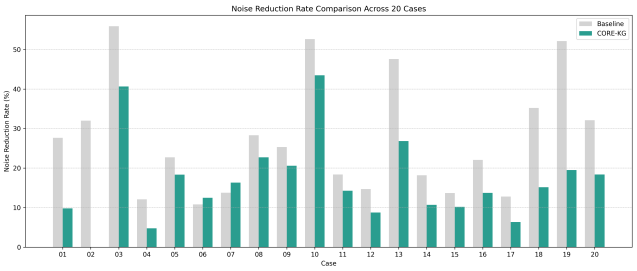

Figure 2: Noise reduction rate comparison between baseline and CORE-KG across 20 legal cases.

Figures 1, and 2 present the case-wise comparison of node duplication rate, and legal noise rate between the baseline GraphRAG system and our CORE-KG pipeline across all 20 legal cases.

Figure 1 presents a case-wise comparison of node duplication rates between the baseline and CORE-KG across 20 legal cases. CORE-KG reduces duplication in all 20 cases, often with substantial improvements. Notably, Cases 2, 6, and 14 exhibit the most significant absolute reductions—approximately 16%, 33.51%, and 20%, respectively—demonstrating CORE-KG's effectiveness in resolving entities with high surface variability. Case 06 shows the largest improvement, reducing the duplication rate from 39.76% to 6.25%, due to successful consolidation of highly variable entity mentions. Case 20, while still exhibiting high duplication for both methods (45.68% baseline vs. 44.90% CORE-KG), remains the most challenging, reflecting limitations under extreme entity density and naming ambiguity. In a few cases, CORE-KG's duplication rate appears close to the baseline—this occurs when the total number of entities and duplicate nodes is significantly reduced in the CORE-KG output. When the absolute number of duplicates is already low, even a small number of remaining duplicates can result in a higher relative percentage. Overall, CORE-KG achieves a relative reduction of 33.58%, lowering the average duplication rate from 30.38% to 20.27%.

Figure 2 presents a comparison of legal noise rates across 20 cases. CORE-KG consistently achieves lower noise rates than the baseline, with an average reduction from 27.41% to 16.89%, yielding a relative improvement of 38.37%. The largest gains are observed in Cases 2, 13, and 19, where the baseline graphs contain heavy legal boilerplate—such as court procedures, appeal details, and legal standards—that CORE-KG successfully suppresses. For example, in Case 2, noise drops from 32.00% to 0.00%, while Case 13 shows a reduction from 47.57% to 26.83%, and Case 19 from 52.13% to 19.51%. In fact, for Case 2, the noise reduction is particularly sharp, bringing it down to zero. There are two cases where the performance of the baseline was better than CORE-KG—specifically, Cases 6 and 7. This is likely due to the limited presence of extractable procedural entities in these cases, which reduces the impact of CORE-KG's in-prompt filtering mechanism. Additionally, because CORE-KG significantly reduces the total number of extracted entities through aggressive consolidation and filtering, even a small number of residual noisy nodes can result in a proportionally higher noise percentage. This effect is particularly visible when the overall entity count is low, leading to skewed ratios despite CORE-KG's improved absolute performance.

Overall, these case-wise breakdowns reinforce the robustness of CORE-KG. While Table 1 summarizes the average gains, these figures demonstrate that CORE-KG achieves consistent structural improvements across varied legal case narratives, and noise without sacrificing contextual accuracy.

B ENTITY AND RELATIONSHIP EXTRACTION PROMPT

Figure 3 presents the full prompt used in our pipeline to guide the large language model (LLM) for structured extraction of entities and relationships from legal case documents. This prompt corresponds to the description in Section 3.3.1 and reflects several task-specific strategies designed to reduce noise and improve precision in knowledge graph construction.

The prompt explicitly defines the seven targeted entity types relevant to human smuggling analysis and provides extraction instructions in a fixed order to reduce attention diffusion. It also incorporates filtering instructions to exclude high-frequency legal or government-related entities that are not structurally important to smuggling networks. In addition, each extracted relationship is assigned a confidence score based on contextual clarity, and a structured output format ensures consistent parsing.

Few-shot examples are included at the end of the prompt to illustrate the expected input-output structure. These examples help reinforce output formatting, relationship modeling, and entity-type mapping.

This prompt serves as a core component of our extraction pipeline, and its design contributes directly to the improvements reported in Section 5.2.1.

LLM Prompt for Entity and Relationship Extraction

-Goal-
You are an expert in Named Entity and Relationship Extraction (NER-RE) for legal case documents related to human smuggling. Your task is to extract only the specified entity types [entity_types] and explicit relationships between them, without inference or completion. These outputs will be used to construct a Knowledge Graph for analyzing smuggling networks. You will receive entity definitions, input text, and structured examples—study them carefully before extraction to ensure strict factual accuracy.
Do not extract entities corresponding to governmental organizations or entities closely related to the trial, criminal law and law procedures (e.g., jury, government, court, prosecution, etc.). These are out of scope.

-Entity Type Definitions-
1. **PERSON**: Any individual's name, including smugglers, agents, and undocumented migrants.
2. **LOCATION**: Geographical areas (e.g., city, state, country).
3. **ORGANIZATION**: Smuggling rings, drug cartels, and other formal groups.
4. **MEANS_OF_TRANSPORTATION**: Vehicles like car, truck, 18-wheeler.
5. **MEANS_OF_COMMUNICATION**: Tools like phone, WhatsApp.
6. **ROUTES**: Roads, highways, or freeways used in smuggling.
7. **SMUGGLED_ITEMS**: Goods like drugs, weapons, or undocumented aliens.

-Steps-

(1) **Entity Extraction:** Extract only explicitly stated entities of type `[entity_types]`. Do not infer or complete missing information. For each, extract the following fields: `entity_name` — Capitalized name as it appears.
`entity_type` — One of: `[entity_types]`
`entity_description` — Detailed description of the entity's role or attributes.

Do not extract any entities related to government or legal proceedings (e.g., court, jury, prosecution, law enforcement, etc.).

Extract entity types in the following order:
PERSON: If a person appears with a title (e.g., "Agent Ryan Lewis"), extract only the full name (e.g., "Ryan Lewis") as the `entity_name` and include the title in `entity_description`.
LOCATION: Combine city and state into a single entity (e.g., `Laredo, Texas`).
MEANS_OF_TRANSPORTATION, MEANS_OF_COMMUNICATION, ROUTES, SMUGGLED_ITEMS, ORGANIZATION: Extract as relevant.

Format each entity as:
`("entity"{tuple_delimiter}entity_name{tuple_delimiter}`
`entity_type{tuple_delimiter}entity_description")`

(2) **Relationship Extraction:** From the entities identified, extract all clearly stated relationships, even if embedded in complex sentences.
For each relationship, extract:
`source_entity` — Source entity from step 1
`target_entity` — Target entity from step 1
`relationship_description` — Explanation of the connection
`relationship_strength` — Score between 0–10:
- **0–3 (Weak):** Indirect or uncertain (e.g., "may have…")
- **4–6 (Moderate):** Explicit but lacks strong context
- **7–10 (Strong):** Clear, direct, and contextually supported

Format each relationship as:
`("relationship"{tuple_delimiter}source_entity{tuple_delimiter}`
`target_entity{tuple_delimiter}relationship_description{tuple_delimiter}`
`relationship_strength")`

(3) **Filter Government Entities:** If any government-related entities or relationships are mistakenly extracted, remove them.
(4) **Output Format:** Return all extracted entities and relationships as a single list using `{record_delimiter}` as the separator.
(5) **Completion Token:** End the output with: `{completion_delimiter}`

- Few-shot Examples-
Example 1
Input: `Smugglers from the Horizon Smuggling Ring used WhatsApp.`
Output:
`("entity"{tuple_delimiter}SMUGGLERS{tuple_delimiter}PERSON{tuple_delimiter}...)` {record_delimiter}
`("entity"{tuple_delimiter}WHATSAPP{tuple_delimiter}MEANS_OF_COMMUNICATION{tuple_delimiter}...)`
{record_delimiter}
`("relationship"{tuple_delimiter}SMUGGLERS{tuple_delimiter}WHATSAPP{tuple_delimiter}...)`
{record_delimiter}
`{completion_delimiter}`
...

Figure 3: Prompt used for entity and relationship extraction from case documents.

C COMPARISON OF GRAPH QUALITY FOR A REPRESENTATIVE LEGAL CASE

Figures 4 and 5 present a side-by-side comparison of the knowledge graphs generated by the baseline GraphRAG system and our proposed CORE-KG pipeline for the same input case 2.

To improve interpretability, relationship labels have been omitted and node sizes are scaled by their degree (i.e., the number of connections each entity participates in), which serves as a visual proxy for node centrality.

Figure 5 illustrates the baseline knowledge graph generated using GraphRAG for Case 2. The graph contains several redundant nodes, including surface-level variants of the same entity, highlighted using colored rectangles. The same color is used to indicate duplicate nodes that refer to the same entity expressed in different forms. The baseline system redundantly represents the main actor, `A.Y.`, as multiple separate nodes such as `Y.` and `A.Y.`. These entities appear disconnected or only indirectly linked, fragmenting the narrative and inflating the graph's structural complexity. Additionally, legal and procedural terms such as `Court`, `Defense`, and `Court of Appeals` dominate the graph, contributing to legal noise despite being irrelevant to the underlying smuggling network structure.

In contrast, Figure 4 shows CORE-KG correctly merging the coreferent mentions of the main actor into a single `A.Y.` node. Other coreferences have also been resolved and resultant into clean and more coherent graph. This result into reduction in the total node count from 86 in baseline graph to 42 in core-kg graph. Moreover the relationship count has reduced from 117 in the baseline graph to the 70 in the CORE-KG graph. The result is a cleaner, more coherent structure that better represents the underlying smuggling narrative. Government-related entities are filtered out during prompt-guided extraction, reducing noise and improving analytical clarity.

These visualizations support the quantitative results in Table 1 and the qualitative analysis in Section 5.2.1, demonstrating that CORE-KG produces graphs that are both semantically aligned and structurally faithful to the case narrative.

D COREFERENCE RESOLUTION PROMPT — PERSON ENTITY TYPE

Appendix Figure 6 displays the full prompt used for resolving coreferences related to the `Person` entity type. This prompt is a central component of our type-aware coreference resolution strategy, described in Section 3.2.2.

The design reflects our structured approach, with components including a persona definition, task descripton, contextual information, entity-type-specific resolution rules, and few-shot examples tailored to the legal domain. The prompt emphasizes strict text preservation, title removal, and disambiguation logic when multiple persons share surnames. It also includes guidance for resolving both singular and plural references like "the defendant" and "the defendants." The few-shot examples at the end of the prompt illustrate how the system should resolve references across varying contexts—capturing patterns such as title abbreviation (e.g., BPA Samson), full title references (e.g., Border Patrol Agent Gabriel Luis Rivera Dominguez), and compound name disambiguation. This structured design helps ensure consistency, accuracy, and generalization across diverse legal case documents in our dataset.

Prompt for Coreference Resolution — Person Entity Type

- Goal -
You are a highly precise and intelligent coreference resolution system designed to support named entity recognition (NER) and knowledge graph construction. Your task is to resolve all coreferences related to the Person entity type—including roles and titles (e.g., Defendant, Officer, Agent)—in a given input text, while strictly preserving its original structure and wording. The resolved output will be used for extracting person entities and relationships in the context of human smuggling networks. Therefore, maintaining accuracy and consistency is critical. Do not summarize, explain, or alter the text—only return the full, unmodified input with Person coreferences resolved according to the rules below.
Note: This is an unsupervised coreference resolution task. The instructions are designed to guide you in resolving person-related references. While examples are provided, they do not cover all scenarios. You must infer and apply coreference logic based on contextual understanding, even when phrasing or structure varies. **- Coreference Resolution Rules — Person Entity Type -**

- After a person is introduced with their full name (e.g., Marco Reyes), replace all subsequent mentions—including last name only (e.g., Reyes), role + last name (e.g., Agent Reyes), and abbreviated forms (e.g., BPA Reyes)—with the full name only.
- In all coreference resolutions, strip titles from mentions. For example, "Agent Dominguez" or "Agent Diego Manuel Cruz Herrera" should resolve to "Hector Luis Rivera Dominguez" or "Diego Manuel Cruz Herrera".
- For compound names, match based on the final component (e.g., Dominguez, Lewis) and resolve to the full name.
- If two or more individuals share a last name, resolve ambiguous mentions conservatively—default to the most recently introduced full name unless context clearly indicates otherwise.
- If abbreviated titles appear (e.g., BPA, Agent, Officer + Last Name), remove the title and resolve to the full name.
- If a person is introduced as "Defendant Elena Sofia Rivera Delgado", resolve it to "Elena Sofia Rivera Delgado" immediately and throughout.
- If someone is introduced as "Border Patrol Agent Gabriel Luis Rivera Dominguez", retain this in the first mention, but resolve all later mentions (e.g., "Agent Dominguez") to "Gabriel Luis Rivera Dominguez".
- Apply all replacements across the entire document, including headers, transcripts, footnotes, and end-of-document text.

Multiple Defendants:

- If multiple defendants are introduced, resolve "the defendants" to a comma-separated list of their full names, in the order introduced.
- "The defendant" (singular) should resolve to the most recently mentioned full defendant name unless context indicates otherwise.
- Always resolve all such role-based mentions, even in peripheral document sections.

- Examples -
Example 1:
Input: Border Patrol Agent Dean Samson observed the vehicle. BPA Samson contacted another agent.
Output: Border Patrol Agent Dean Samson observed the vehicle. Dean Samson contacted another agent.
Example 2:
Input: Border Patrol Agent ictor Manuel Rojas Delgado led the operation. Delgado coordinated with the local sheriff.
Output: Border Patrol Agent Victor Manuel Rojas Delgado led the operation. Victor Manuel Rojas Delgado coordinated with the local sheriff. ....

- Input Text -
Resolve all Person entity coreferences in the following document, including those in footnotes and headers. Return only the modified text. If none exist, return the input unchanged. Do not summarize or explain. `Input_text: {input_text}`
`Output:`

Figure 6: Prompt used for resolving coreferences related to Person entities. This includes resolution rules, behavioral constraints, and illustrative examples.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009

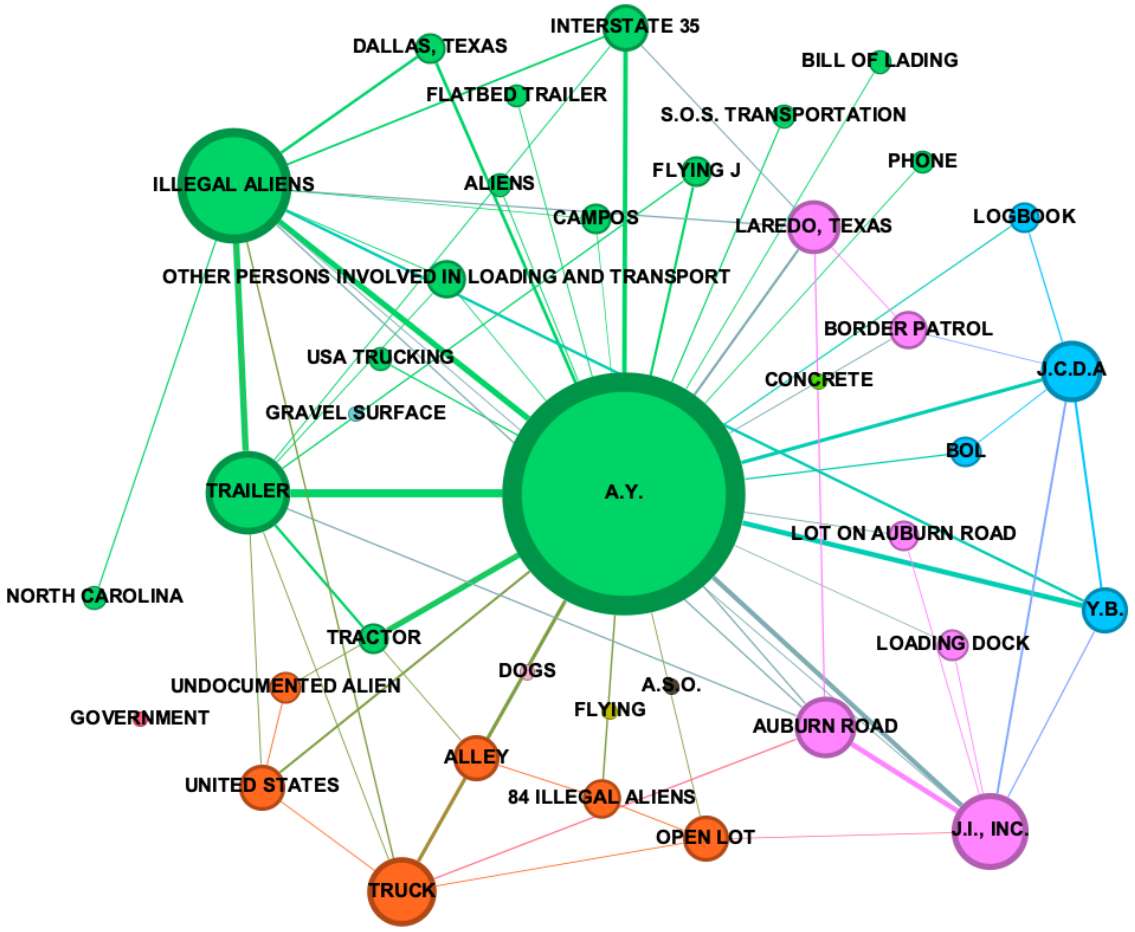

Figure 4: Knowledge graph generated by CORE-KG for a representative legal case. The graph demonstrates resolved coreference, improved coherence, reduced legal noise, and more precise entity linking compared to the baseline output.

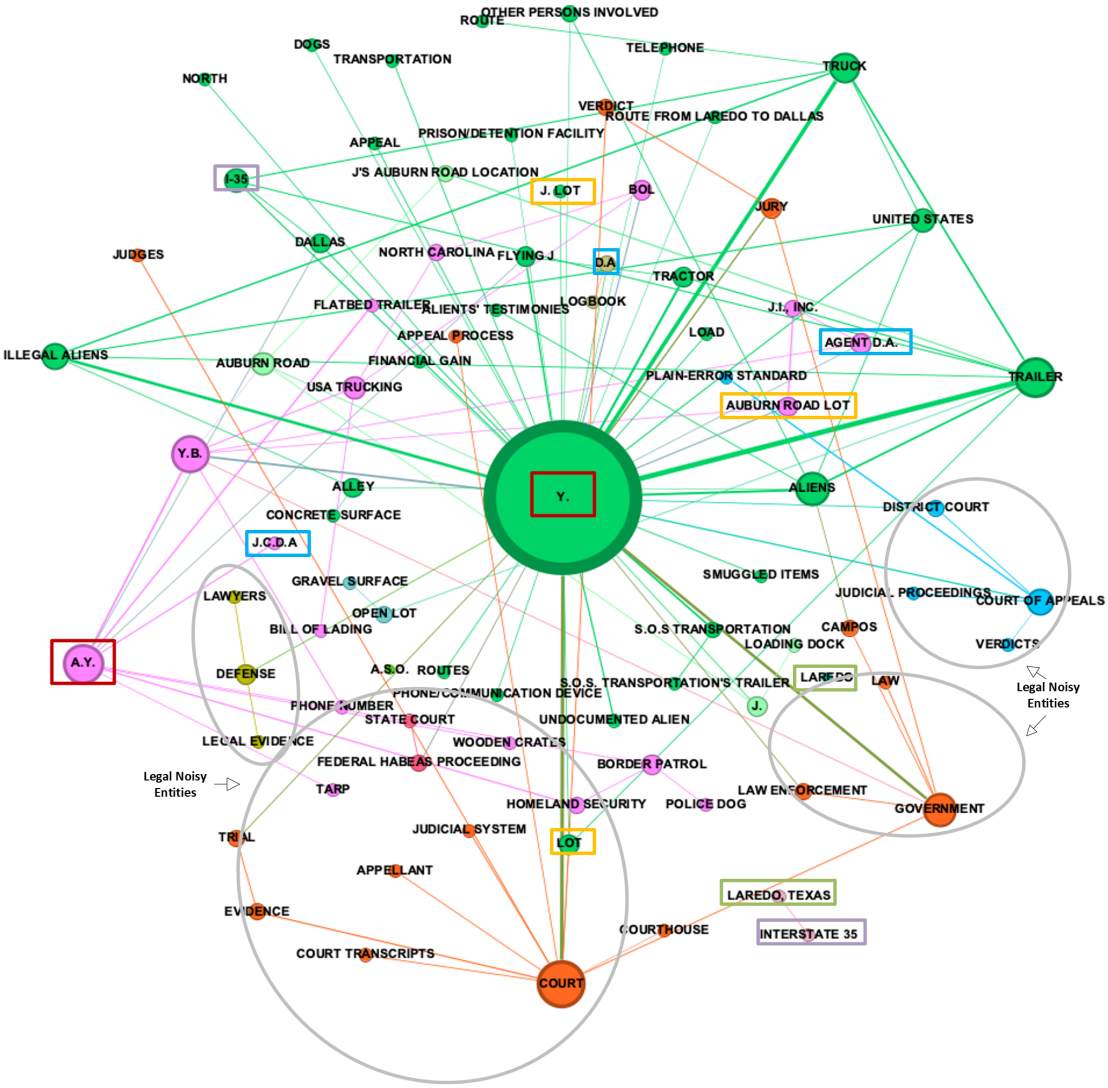

Figure 5: Baseline knowledge graph generated using GraphRAG for a representative legal case. The graph contains several redundant nodes, generic entities, and visually dense connections, which reduce overall clarity. Key duplicate nodes are highlighted using rectangles of the same color, indicating repeated entities that fragment the graph structure.