LLM-Assisted Triple Extraction from Historical Texts

VLAD-ANDREI ALEXE, University of Twente, The Netherlands

Historical texts are rich sources of knowledge, but their unstructured nature makes it difficult to analyze them systematically. This research explores how Large Language Models (LLMs) can be used to extract subject-predicate-object (SPO) triples from historical narratives, enabling the construction of structured knowledge graphs (KGs). Focusing on three representative LLM-based frameworks (AiKG, Triplex, and GPT o3), we assess their effectiveness in handling the linguistic complexity, archaic vocabulary, and contextual nuance found in historical documents. The study includes both quantitative evaluations (e.g., precision, recall, F1) and qualitative assessments of factual accuracy, completeness, faithfulness, and entity alignment. Our findings highlight significant variation in performance across frameworks, with GPT o3 demonstrating the best balance of coverage and semantic accuracy. This work contributes to the growing field of digital humanities by showing how LLMs can support historical research through the automated extraction of structured information, while also identifying current limitations and areas for improvement in LLM-based extraction tools.

Additional Key Words and Phrases: Large Language Models, Knowledge Graphs, Triple Extraction, Historical Texts, NLP

1. Introduction

History is a rich domain filled with interconnected events, figures, locations and dates. Gaining knowledge in this area can help us understand patterns of the past that otherwise would be difficult to see in the present [1]. However, most historical knowledge is locked in unstructured narratives, such as textbooks, academic articles and encyclopedic entries, which makes it difficult to visualize, analyze and query [2].

Extracting structured representations in the form of subject-predicate-object (SPO) triples from these types of text offers a way to turn complex narratives into machine readable knowledge graphs (KG) [3]. In turn, KGs could facilitate a better understanding of historical texts, by enhancing comprehension through the visual representation of entities and their relationships, leading to deeper insights. [4].

Recent advances in artificial intelligence (AI), particularly in natural language processing (NLP), have significantly improved the automated extraction and structuring of knowledge from unstructured texts. Within AI, machine learning (ML) [5] enables systems to improve from data without explicit programming. Large Language Models (LLMs), a recent advancement in ML, are trained on vast text-based datasets to understand and generate human language, significantly enhancing the performance of NLP tasks [6].

The goal of this research paper is to provide a comparison of existing frameworks that leverage LLMs to improve subject-predicate-object triple extraction from unstructured texts, with the focus being on their applicability to historical texts.

TScIT 43, July 4, 2025, Enschede, The Netherlands

2. Problem Statement

Recent advancements in artificial intelligence, particularly the development of LLMs, offer relevant options for automating the extraction of information from unstructured texts. LLMs have demonstrated capabilities in understanding and generating human-like text, suggesting potential for their application in extracting SPO triples from historical narratives. However, the effectiveness of LLMs in this specific context remains underexplored. Historical texts often contain complex language, archaic terms, and nuanced contexts that pose challenges for automated extraction methods. Moreover, this research could be meaningful in departments such as education and research. For example, in educational settings, these structured representations can serve as valuable tools for teaching and learning. They can help students visualize connections between historical figures, events, and places, potentially leading to higher engagement and better comprehension. By evaluating existing frameworks and methodologies that apply LLMs to information extraction, the study seeks to identify best practices and potential limitations in the context of historical document analysis.

3. Research Question

The problem statement leads to the overarching research question:

How can Large Language Models be effectively applied to extract SPO triples from historical texts?

This research question can be answered with the aid of the following sub-questions:

- **sRQ1:** What are the current frameworks that employ LLMs for SPO triple extraction?
- **sRQ2:** How do these LLM-based frameworks perform in extracting SPO triples when applied to historical texts?
- **sRQ3:** What are the limitations of using LLMs for SPO triple extraction in historical contexts?

4. Related Work

Triple extraction, the identification of subject–predicate–object structures within unstructured text, has been a foundational step in building knowledge graphs (KGs) for a long time now [7]. Traditionally, it relied on rule-based systems or supervised relation extraction models, which often require extensive domain-specific tuning and annotated texts. While effective in structured settings, such methods struggle with generalization [8]. Knowledge graphs, once constructed, are essential tools for semantic search, reasoning, and question answering [9]. In recent years, large language models (LLMs) have emerged as highly capable engines for triple extraction, offering greater adaptability and semantic understanding [6]. This

section surveys recent LLM-driven approaches to triple extraction, grouped into prompt-based, fine-tuned, and hybrid methods.

A first class of frameworks employs **prompt-based extraction**, leveraging the natural language capabilities of LLMs without requiring additional training. The GPT o3 model [10], representing commercial LLMs like GPT-3.5 and GPT-4, is used in this study as a baseline through a carefully designed prompt that directly request structured SPO triples. In a similar vein, KG-LLM-Prompting [11] provides a minimalist prompting pipeline that combines named entity recognition and relation prediction into labeled triples using zero-shot LLM inference. A more advanced example is SPIRES, part of the OntoGPT project [12], which introduces structured prompting to query LLMs for schema-compliant extractions. It recursively interrogates the text with predefined ontological classes, grounding entities to unique identifiers and using reasoning mechanisms to validate output. SPIRES has demonstrated success in extracting richly structured knowledge in domains like biomedical mechanisms and recipe analysis.

The second group consists of fine-tuned language models, where the LLM itself is adapted for the structured task. Triplex [13], hosted on Hugging Face, is a finetuned variant of Phi-3.8B—an instruction-following small language model developed by Microsoft. It is specifically trained on DBpedia- and Wikidata-style corpora to output SPO triples natively. This makes it suitable for efficient, schema-consistent extraction without the need for elaborate prompt engineering. Similarly, ICKG [14] is an instruction-tuned model that specializes in generating triples and aligning them with existing knowledge bases. Though documentation is limited, its design emphasizes fine control over structure and relation grounding.

The final set of frameworks uses **hybrid pipelines**, combining LLM-based extraction with post-processing, visualization, or additional symbolic methods. EDC (Extract–Define–Canonicalize) [15] proposes a three-stage architecture that begins with open information extraction, then induces a schema from patterns, and finally canonicalizes entities, each step augmented by LLM-generated suggestions. LocalKnowledgeGraphExtraction [16] supports similar triple extraction workflows but executes entirely on local hardware, preserving privacy by avoiding remote API calls. DeepKE [17], a modular toolkit originally intended for neural relation extraction, has been extended to support LLM-enhanced entity and attribute extraction, providing flexibility for document-level use cases. Several tools integrate triple extraction with immediate downstream use. The AI Powered Knowledge Graph Generator (AiKG) [18] pairs LLM output with dynamic graph construction, while kg-gen [19] facilitates relationship mapping and retrieval-augmented generation pipelines. Knowledge Graph Builder [20], developed by Neo4j Labs, translates text into structured graphs compatible with graph databases like Neo4j via LangChain integration. KGViz [21] offers a local graphical frontend for browsing extracted triples interactively. Finally, llmgraph [22] focuses on extracting triples from Wikipedia articles using ChatGPT and exporting the results in formats suitable for HTML visualization or GraphML analysis.

Together, these frameworks highlight the breadth of approaches made possible by LLMs. Prompt-based tools prioritize flexibility and low setup costs but can be inconsistent. Fine-tuned models deliver structured, reliable output with higher precision. Hybrid frameworks combine LLM reasoning with structured pipelines to support scalable, interactive KG construction. Across all designs, LLMs serve as the core component that bridges unstructured input with semantically meaningful structure.

5. Methods of Research

In order to systematically evaluate LLM-based SPO triple extraction frameworks on historical texts, we adopted a multi-stage methodology. First, we conducted a literature survey and tool inventory to identify candidate frameworks that leverage large language models for knowledge graph construction. From this pool, we selected a representative subset of frameworks based on different criteria.

Next, we curated a set of three historical documents and extracted focused excerpts that contained rich subject–predicate–object relations. Each framework was then applied to these excerpts under comparable settings, producing raw sets of predicted triples. The raw outputs were subjected to a through cleaning and normalization.

We implemented a custom Python script to compute evaluation metrics—precision, recall, F1, and omission rates—using a strict similarity threshold (0.75) and a lenient "possible-match" threshold (0.50). The script also reports matched, unmatched, and possibly-matched pairs to facilitate both quantitative analysis and qualitative inspection. This combined automated and manual review process ensures that we capture not only the numerical performance of each framework but also its behavior on real historical passages.

5.1 Frameworks Selection Criteria

The framework selection criteria were based on four practical considerations. First, tools had to be freely available or low-cost to ensure accessibility. Second, we required open-source implementations hosted in public repositories (e.g., GitHub or Hugging Face) to guarantee reproducibility and extensibility. Third, each framework needed to offer a clear, straightforward installation process, complete with setup instructions, dependencies, or packaged releases, so that it could be deployed with an appropriate level of effort, because of the time constraints. Finally, every candidate had to integrate at least one large language model into its extraction pipeline, confirming its use of LLM capabilities. These criteria ensured our evaluation focused on tools that are both feasible to deploy and representative of the current LLM-driven SPO extraction landscape. Due to limited number of frameworks that fit the criteria, we also included a few less-than-ideal cases to provide a broader perspective on available options.

Ultimately, due to time constraints, only three out of thirteen frameworks were picked for continuing the experiment. The choice was made based on the availability of the resource, how well the frameworks fit our selection criteria, how manageable the setup process was, and lastly, based on a test extraction from a small sample of a historical text.

5.2 Source Texts and Excerpt Selection

To construct a meaningful evaluation dataset, the selected texts had to meet several key criteria. First, all source materials were required to be freely available and about historical events, ideally written within a semi-academic or academic context. This ensured both accessibility and a reasonable level of factual reliability. Moreover, temporal diversity was an essential factor in the selection process. The texts were chosen to reflect distinct historical periods (e.g., Roman Empire, French Revolution, and World War II), thus broadening the scope of content types and challenges encountered in knowledge extraction.

The excerpts themselves were deliberately curated to vary in narrative structure and complexity. The Roman Empire text [23] is more narrative-driven and was written in a differrent time-period (initially, but was revised a few time since then), the French Revolution [24] excerpt offers a balance between narrative and academic tone, while the World War II text [25] is densely packed with named entities and organizational references, making it especially challenging in terms of entity alignment and relation disambiguation. This variation ensures a more comprehensive evaluation of each framework's strengths and limitations. The text excerpts are about 500 words in length, with a 5% margin. This limit was chosen deliberately to ensure that the manual annotation of the texts is feasible in the allotted time. Lastly, upon selection, the excerpts were also subjected to a simple cleaning process, which entails removing any symbols or marks that were present in the text as references or formatting cues. The excerpts can be viewed in *[Appendix B: Text Excerpts]*.

5.3 Conducting the experiment

The experimental procedure began with the manual construction of ground truth sets, also known as the gold standard sets, of SPO triples for each of the three selected historical text excerpts. These reference triples were crafted to reflect the factual information, and at times, inferred meaning from the text excerpts. Each historical text excerpt was manually annotated through close reading and iterative analysis. Key entities and relationships were identified directly from the text, with care taken to preserve factual accuracy and semantic clarity. Only explicitly stated or unambiguously implied information was encoded as (Subject, Predicate, Object) triples. Coreference resolution and disambiguation were performed manually, ensuring consistency in entity representation across triples. This process aimed to form a reliable benchmark for evaluating automated extraction frameworks. You can see the ground truth triples in *[Appendix C: Ground Truth/Gold-standard Triples]*.

Following this, each triple extraction framework was executed independently on all three texts (see *[Appendix E: Prompt Used For GPT o3 Model]* in order to see what prompt was used to extract triples via GPT o3's Model). The resulting outputs were subsequently cleaned and standardized to ensure they adhered to the same structural format as the ground truth triples, thus allowing for consistent and fair comparison.

To evaluate the similarity between the extracted triples and the reference sets, a custom Python-based matching algorithm was developed. This algorithm computes the similarity of each extracted triple to the gold standard based on overlapping character sequences and word order. This approach offers a lightweight way to approximate semantic similarity without requiring deep linguistic analysis. It tolerates minor wording differences while preserving the structural order of triples. A surface-level, character-based method ensured that all evaluations remained grounded in the actual textual content without external interpretation. A triple is considered a true positive match if its similarity score exceeds 0.75 on a scale from 0 to 1. The threshold cannot be too high as it would require strings to be almost identical. The 0.75 limit was selected through a trial-and-error process, with the main criterion being that the algorithm picks up on as many correct matches as possible, while keeping the false positive rate as close to 0 as possible. Moreover, triples scoring between 0.5 and 0.74 are considered partial or fuzzy matches and stored in a separate list for manual inspection. This was particularly helpful in observing whether the frameworks' extraction works as intended, or if the matching is erroneous due to the method of comparing the triples used in the algorithm. Furthermore, two additional lists are generated: one for unmatched gold triples, indicating potential recall failures or missed content, and one for unmatched extracted triples, which are indicative of hallucinated or irrelevant content.

This list-based matching system provides significant insights beyond raw scores, allowing for qualitative assessment of errors. This was done manually and allowed us to closely observe what kinds of errors arise. For instance, it highlights cases where a match should have occurred but did not due to surface-level differences, such as synonymous phrasing, reversed subject-object roles, or stylistic variations in entity names (e.g., "Crim Tartary" and "Chersonesus Taurica"). Such nuances are particularly important in high-performing systems, where the semantic content is often preserved but lexical or syntactic mismatch penalizes the score. Given the scale of the dataset, approximately 150-200 extracted triples and around 50-70 ground-truth triples, **per text**, automation was necessary due to time constraints. Although this method is not flawless and lacks the granularity of human annotation, it provided a reasonable compromise between efficiency and accuracy within the scope of the project timeline.

In terms of metrics, the python algorithm computes standard evaluation scores such as **precision, recall, F1 score, and omission rate (defined as 1.0 - recall)**. These metrics are standard in information extraction tasks because they offer a balanced view of a system's performance. **Precision** measures how many of the extracted triples are correct, reflecting accuracy. **Recall** assesses how many relevant triples were successfully captured, indicating completeness. **F1 score** combines both into a single metric to balance precision and recall, especially useful when there's a trade-off between the two. Omission rate (1.0 – recall), a custom metric, highlights how much information was missed, which is highly relevant in applications where coverage is as important as correctness.

Care was taken throughout the process to fact-check the matching process wherever feasible. This proceeding was necessary in order to avoid incorporating or endorsing misinformation generated by the algorithm's logic and to ensure that the study accurately describes the framework's abilities. Importantly, these quantitative metrics

are treated differently depending on the frameworks' performance. For underperforming frameworks, they serve as reliable indicators of extraction quality due to the significant differences from the ground truth. For high-performing systems, however, these scores represent lower-bound estimates of accuracy, as they fail to account for semantic equivalence, inferential reasoning, or stylistic variation in entity labeling.

Finally, based on the aforementioned produced lists, a qualitative analysis has been made on the behaviour of each framework, on each text. These were compiled in overviews of the individual frameworks' behaviour. The overviews were separated into 4 main dimensions: *Factual Accuracy, Completeness (under-generation), Faithfulness (over-generation), and Entity Alignment & Resolution.* These four dimensions provided enough information to form an educated opinion on what each framework is capable of and what its limitations are.

6. Results

This section presents the quantitative results obtained from evaluating the triple extraction frameworks on the three historical text excerpts. The evaluation focuses on both the volume and quality of the extracted knowledge graph triples, comparing them against hand-crafted gold-standard sets. We report standard information extraction metrics, including precision, recall, F1 score, and omission rate, alongside statistics on triple-level matches and entity-level coverage. These results serve as the empirical basis for the interpretive discussion provided in the subsequent **Results Analysis & Discussion** section.

6.1 Extracted triples

The table below (*Table 1*) summarizes the number of triples extracted by each framework for each of the three text excerpts, alongside the corresponding number of gold-standard (ground truth) triples. This provides an initial sense of the extraction volume and helps contextualize the evaluation metrics presented in later sections.

Table 1. Triple Counts by Framework and Text

Text	Framework	Extracted Triples	Gold-Standard Triples
Roman Empire	AiKG	128	50
	Triplex	10	
	GPT o3	26	
French Revolution	AiKG	122	57
	Triplex	11	
	GPT o3	48	
World War II	AiKG	101	71
	Triplex	12	
	GPT o3	39	

6.2 Qualitative Evaluation Metrics

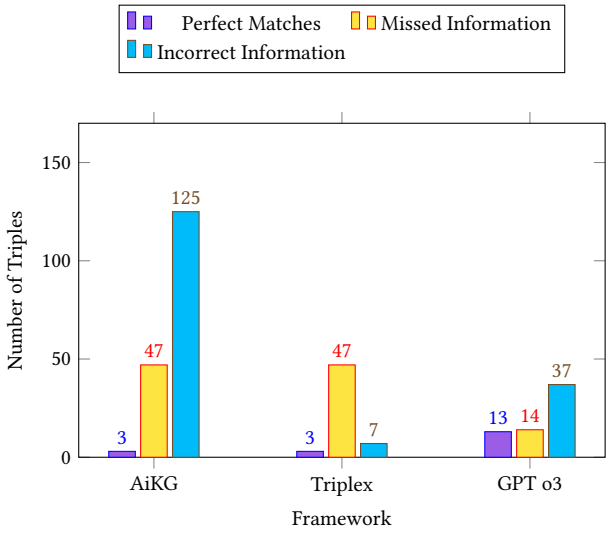

Fig. 1. Triple Matching: Roman Empire

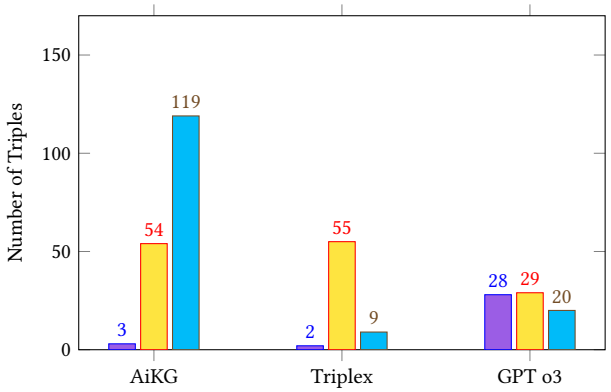

Fig. 2. Triple Matching: French Revolution

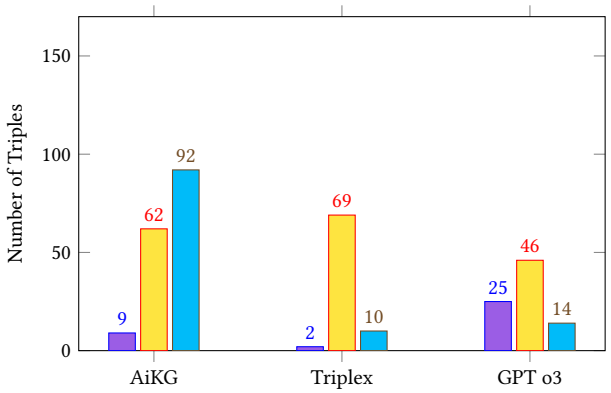

Fig. 3. Triple Matching: World War II

From an objective standpoint, the most desirable outcome is for Perfect Matches to be as high as possible, while the other two categories, Missed Information, and Incorrect Information to be numerically low and tightly clustered across frameworks (see *Fig 1, Fig 2, Fig 3*). Low counts of Missed Information indicate strong recall (few missed facts), while low counts of Incorrect Information reflect high precision (few wrong triples). When the bars for these two error types are small and similar in height, it means each system is extracting a balanced amount of information without overwhelming noise or critical omissions. Together, these three counts help us identify whether a system tends to miss important content (under-generation) or include too much irrelevant or incorrect information (over-generation). This provides a simple but effective way to estimate how accurate and reliable each framework is when used for knowledge extraction tasks.

In addition to these summary statistics, the Python evaluation tool also provided access to all possible (partial) matches (see *[Appendix D: Results - Trimmed Examples]*), which were useful for manual inspection. These cases allowed me to observe when mismatches were caused by limitations in the algorithm's code, for example, small lexical differences or reordered elements, and when they were clearly due to the framework itself. Although this information was not used to alter the core evaluation results, so as to keep the automated comparison consistent and unbiased, it played an important role in the qualitative error analysis. These insights are further discussed in the Results Analysis and Discussion section.

6.3 Quantitative Research Metrics

To objectively assess the extraction performance of each framework, we computed standard information retrieval metrics: precision, recall, F1 score, and a custom omission rate (which is defined as 1.0 - recall). These were calculated using the number of true positives (correctly matched triples), total predicted triples, and gold-standard triples per framework and text. The table below presents the results for all three historical excerpts across the three evaluated frameworks.

Table 2. Precision, Recall, F1 Score, and Omission Rate per Framework and Text. Best score in bold script

Text	Framework	Precision	Recall	F1	Omission Rate
Roman Empire	AiKG	0.023	0.060	0.034	0.940
	Triplex	0.300	0.060	0.100	0.940
	GPT o3	0.482	0.260	0.338	0.740
French Rev.	AiKG	0.025	0.053	0.034	0.947
	Triplex	0.182	0.035	0.059	0.965
	GPT o3	0.583	0.491	0.533	0.509
World War II	AiKG	0.089	0.127	0.105	0.873
	Triplex	0.167	0.028	0.048	0.972
	GPT o3	0.641	0.352	0.455	0.648
Average	AiKG	0.046	0.080	0.058	0.920
	Triplex	0.216	0.041	0.069	0.959
	GPT o3	**0.569**	**0.368**	**0.442**	**0.632**

The selected metrics, precision, recall and F1 score are widely used in information extraction and are particularly well-suited for evaluating triple extraction tasks. Precision captures the accuracy of the extracted triples, indicating how many of the proposed relations are actually correct. Recall measures the system's ability to recover relevant information, showing how much of the ground-truth content was successfully extracted. The F1 score balances these two aspects, offering a single, interpretable measure of overall performance. Omission rate, calculated as the complement of recall, highlights how much information was left out, a critical factor in tasks focused on completeness. Together, these metrics provide a well-rounded view of each framework's behavior by quantifying both over-generation and under-generation. This makes them especially relevant for the goals of this experiment, which seeks not only to identify which frameworks are more accurate, but also to understand the nature and extent of their extraction errors and limitations.

7. Results Analysis & Discussion

This section synthesizes the findings from both the quantitative metrics and the qualitative observations, offering a comparative interpretation of how each framework performed in the extraction of subject-predicate-object triples.

7.1 Comparative Framework Behavior

Each of the three evaluated frameworks demonstrates distinct extraction behavior, evident both from the evaluation metrics and from manual inspection of the outputs.

From this point onward, for the sake of efficiency, **Roman Empire** refers to *The History of the Decline and Fall of the Roman Empire*'s excerpt [23], **French Revolution** refers to *The Oxford History of The French Revolution*'s excerpt [24] and **WW2** refers to *The Second World War*'s excerpt [25].

AiKG tends to over-generate triples, resulting in very low precision scores (e.g., 0.023 for Roman Empire and 0.025 for French Revolution). This behavior is reflected in its consistent production of large numbers of loosely related or incorrect triples. For instance, in **Roman Empire**, the model generated the incorrect triple *(frontier, infested by, East Asia)*, misrepresenting the original text's reference to Germanic tribes. Despite its high output volume, AiKG frequently misses structurally or contextually embedded facts, such as the origins of the National Convention in **French Revolution** or the encirclement of the Ninth Army in **WW2**, leading to omission rates over 0.9 in two of the three texts. This highlights its inconsistent coverage despite high triple counts.

Triplex, by contrast, shows the most conservative extraction strategy. It consistently produced the fewest triples, with omission rates nearing or exceeding 0.95, and precision values that are relatively higher (e.g., 0.3 for Roman Empire), but applied to very limited data. The framework's constrained behavior is primarily due to its design: Triplex requires the user to provide a detailed list of allowed entity types and predicates as part of the prompt. While this schema-driven approach can reduce noise, it also severely restricts recall, as any entity or relation not included in the prompt is effectively invisible to the model. This explains its failure to extract central political actors in **French Revolution** or key military events in **WW2**. Though this is a limitation of the prompt rather than the model itself, it is not

easily avoidable. Improving the results would require substantial manual curation of exhaustive, context-specific lists. Given the time and effort required to tune these lists, frameworks with broader generalization capabilities may offer more efficient alternatives in practice.

GPT o3 achieves a more balanced performance, reflected in moderate to high precision (0.48-0.68) and the highest recall among the frameworks (0.26-0.49). It captures a wide range of facts and tends to preserve the semantic intent of the text, even when phrased differently. For example, it correctly extracted *(Insurrectionary Commune of Paris, was responsible for, fall of the French monarchy)* in **French Revolution**, where other models either missed the fact or generalized it incorrectly. However, its scores are slightly penalized due to surface-level mismatches or rewordings. A few examples are: switching subject and object, or minor naming variations—which leads to underreported recall in automated evaluation. Among the three frameworks, GPT o3 via Prompt Engineeringis the most negatively affected by the limitations of the evaluation code. While the model often produces semantically correct triples, the string-based similarity comparison used in the scoring script fails to recognize meaning-preserving variations. As a result, its actual performance is substantially better than the metrics suggest. In this case, the reported scores should be considered only a lower bound of the model's extraction capability.

Overall, these behaviors illustrate three distinct strategies: AiKG favors aggressive recall at the cost of accuracy, Triplex favors precision at the cost of coverage, and GPT o3 offers a balanced middle ground with the highest semantic alignment despite occasional scoring penalties.

7.2 Limitations of Automated Metrics in Evaluating Extraction Quality

While the chosen evaluation metrics, precision, recall, F1 score, and omission rate, paint a useful quantitative picture, they do not equally reflect the true capabilities of each framework. For low-performing systems such as Triplex and AiKG, these metrics are largely accurate: their low recall values correspond to frequent omissions, and their low or inconsistent precision reflects the presence of hallucinated or fragmented triples. In these cases, the metrics offer a fair and direct measure of performance, highlighting clear limitations in coverage, alignment, or factual grounding.

However, for better-performing systems like GPT o3, the metrics underrepresent actual performance. This is due to the evaluation code's reliance on character and token-level similarity to determine matches between predicted and gold-standard triples. When the model expresses a relation using different wording, switches subject and object order, or names entities slightly differently, the triple may be penalized, even if it correctly conveys the same meaning. For example, the gold triple *(Bosphorus, lost to, Roman arms)* was not counted as a match for the extracted *(Kingdom of Bosphorus, sunk under, Roman arms)*, despite capturing the same historical claim. These kinds of mismatches are frequent in GPT o3's output, where semantic understanding often surpasses surface alignment.

Because of this, the scores reported for GPT o3 should be interpreted as lower-bound estimates. The system's true extraction ability, especially in terms of meaning preservation and contextual understanding, is stronger than the raw metrics suggest. This distinction reinforces the importance of combining quantitative results with qualitative analysis when evaluating high-capacity LLM-based frameworks.

Nonetheless, despite their limitations, the metrics remain highly relevant to the experiment. They provide a consistent and reproducible method for comparing systems, and offer clear insights, particularly for identifying over-generation, under-generation, and major coverage gaps. In the absence of full semantic evaluation, they act as practical indicators of relative performance, especially useful when frameworks diverge widely in behavior. Used together with manual inspection, these metrics form a balanced evaluation strategy that is both systematic and informative.

7.3 Framework-Specific Limitations

This section outlines the key limitations identified for each extraction framework, based on both metric patterns and manual analysis. These limitations reveal not only how each model fails but also why, helping to explain their behavior under different textual and structural conditions.

AiKG

The AiKG framework exhibits a high tendency toward over-generation, producing large numbers of triples with minimal filtering or contextual grounding. This behavior leads to extremely low precision and recall scores, as the model frequently introduces relations that are not factually supported by the source. Many of these triples stem from speculative inference, where loosely connected entities are linked by inferred relationships not explicitly stated in the text. For example, in **Roman Empire**, AiKG generated *(frontier, infested by, East Asia)*, a clear misreading of the source, which referenced nomadic tribes from the Danube region. Additionally, AiKG often fragments or duplicates information, producing multiple low-quality variants of the same relation.

Another limitation lies in entity generalization and substitution. Specific names and roles are frequently replaced with vague labels such as "nomadic tribes" or "men," which weakens the factual clarity of the output. In several cases, such as the handling of Brissot and Vergniaud in **French Revolution**, named individuals are either omitted or merged under broader categories. Overall, AiKG struggles with semantic precision, resulting in noisy, inconsistent extractions that require significant post-processing to become useful.

Triplex

Triplex's performance is constrained by its dependency on strict prompt templates, which require users to predefine exhaustive lists of entity types and predicates. While this makes the output consistent, it also severely restricts the model's flexibility and recall. Any entity or relation not explicitly included in the prompt is simply excluded from extraction. This design choice led to omission rates

as high as 0.97, with many obvious relations left unprocessed, even when clearly stated in the text.

Furthermore, Triplex appears to rely on strict syntactic cues. If relations are not expressed in straightforward subject-verb-object constructions, they are often ignored. For example take the following excerpt: "The very idea of a national convention to give france a republican constitution also originated in the paris sections.". The phrase is quite complex, but the core meaning could be extracted as *(idea of a national convention to give france a republican constitution, originated in, paris sections)*, which was not identified by triplex under any form. This aspect made it ill-equipped to handle complex sentence structures, implied relations, or historically nuanced references. Although the framework does avoid hallucinations, its extreme conservatism makes it unsuitable for broad-scope or semantically rich extraction tasks unless highly customized prompts are developed, an effort that may outweigh the practical benefits in most real-world scenarios.

GPT o3 (via Prompt Engineering)

The GPT o3 model demonstrates strong semantic understanding but is frequently penalized by the evaluation setup. Its primary limitation lies in its mismatch with the evaluation method, which is based on surface-level string similarity. The model often captures the correct meaning but expresses it using paraphrased predicates, reversed entity order, or varying entity formulations. For example, in **WW2**, the gold triple *(Bosphorus, lost to, Roman arms)* was extracted as *(Kingdom of Bosphorus, sunk under, Roman arms)*, which is semantically equivalent but was not counted as a match.

Additionally, GPT o3 occasionally introduces inferred knowledge that, while plausible, goes slightly beyond the literal text, bordering on mild hallucination. As an example, take the gold standard triple *(Kings of Bosphorus, guarded against, plunderers of Sarmatia)* and the extracted triple *(Kings of Bosphorus, guarded, access of country commanding Euxine Sea and Asia Minor)*. The extracted triple conveys the same core idea but is constructed from entirely inferred phrasing, as no such explicit line exists in the source text. Even though that the information is correct, this is still categorized as a hallucination, albeit a low-risk one. Moreover, these cases are relatively rare and often still thematically coherent. The model also shows occasional inconsistency in entity naming (e.g., title vs. proper name), which can lead to further penalties under strict evaluation. Despite these minor issues, GPT o3 remains the most capable framework overall. Its limitations are more a reflection of the constraints of the evaluation pipeline than of any fundamental flaw in the model itself.

7.4 Error types

Throughout the evaluation process, several recurring error types were identified across the different frameworks. These errors varied in frequency and severity depending on the system but were generally aligned with the extraction behavior described earlier. Below is a summary of the most common error types, each accompanied by an illustrative example drawn from the experiment.

- **Surface variation (paraphrase mismatch)**: The extracted triple expresses the same meaning as the gold triple but is worded differently, leading to a failed match.
  Gold: (Bosphorus, lost to, Roman arms)
  Extracted: (Kingdom of Bosphorus, sunk under, Roman arms)
- **Subject-object inversion**: The entities are correct, but their roles are reversed.
  Gold: (Roman Empire, defeated, Germanic tribes)
  Extracted: (Germanic tribes, defeated, Roman Empire)
- **Over-specific or under-specific naming**: Entity mentions diverge in granularity or formulation.
  Gold: (National Convention, originated from, Paris sections)
  Extracted: (Commune, originated from, Paris)
- **Spurious triples (hallucination)**: The system invents a relationship or fact that is not supported by the text.
  Extracted: (Frontier of Danube, was infested by, East Asia)
- **Entity substitution or confusion**: The model replaces the correct entity with an unrelated or misread one.
  Gold: (Santerre, led, National Guard)
  Extracted: (Sansculottes, massacred, Parisian people)

These errors represent key challenges in automated knowledge extraction and underscore the importance of using both quantitative scoring and manual inspection when evaluating system performance.

8. Conclusion

This section represents the culmination of this research. Here we present the answer to all of our research questions, and the key takeaways that we learnt during this study.

8.1 Addressing the Research Questions

This section revisits the research question and its sub-questions in light of the findings presented throughout this thesis.

sRQ1: What are the current frameworks that employ LLMs for SPO triple extraction?

A structured literature and tooling review identified thirteen LLM-based triple extraction frameworks, ranging from minimal-instruction tools to ontology-grounded or pipeline-based systems. These included tools like SPIRES, DeepKe, Triplex, and several LLM-integrated extractors. From this pool, three representative frameworks, AiKG, Triplex, and GPT o3, were selected for deeper experimental evaluation based on their accessibility, complexity, and LLM integration.

sRQ2: How do these LLM-based frameworks perform in extracting SPO triples when applied to historical texts?

Performance varied widely. AiKG extracted the most triples but scored lowest in precision and factual correctness, often introducing speculative or fragmented information. Triplex maintained high syntactic precision but failed to identify most relevant relations due to the strictness of its prompt schema. GPT o3 provided the most

accurate and semantically faithful extractions, though it was penalized by surface-level evaluation limitations. Overall, GPT o3 proved to be the most balanced and adaptable framework for extracting SPO triples from historically rich and stylistically diverse texts.

sRQ3: What are the limitations of using LLMs for SPO triple extraction in historical contexts?

The experiment revealed several limitations that affect how LLM-based frameworks perform in historical domains, many of which stem from how each framework handles language complexity, context, and structure.

One major limitation is over-generation, most evident in AiKG, which frequently produces speculative or fragmented triples that are not fully grounded in the source text. This behavior introduces noise and lowers precision, especially in texts with ambiguous phrasing or loosely connected entities. On the opposite end, Triplex suffers from under-generation. Its dependence on strict prompt templates and predefined schema leads to the exclusion of many valid relations not explicitly covered by the input configuration. This makes it ill-suited for capturing the richness of historical narratives unless the prompt is exhaustively tailored, a task that is time-consuming and resource-intensive.

Another common limitation involves handling of entities and relation structure. Several frameworks misassign subject and object roles, generalize named entities into vague categories, or fail to align different references to the same entity. These errors are particularly problematic in historical texts, where actors and places are often mentioned under multiple names or titles (e.g., "Chersonesus Taurica" vs. "Crim Tartary"). Triples can also diverge in granularity or precision, which reduces factual clarity.

Although some mismatches in evaluation stem from superficial differences in wording, most extraction errors originate within the frameworks themselves. Challenges such as recognizing implied relations, resolving coreference, and navigating complex sentence structures all contribute to missed or malformed triples. These limitations reflect the difficulty of applying LLMs to historical data, which often involves dense language, embedded meaning, and inconsistent terminology.

Overall, LLM-based frameworks still face significant challenges in achieving accurate and complete information extraction from historical texts. The source of these limitations is less about the capacity of the LLMs and more about how each framework constrains or channels that capacity, whether through overgeneralization, rigid design, or limited contextual awareness.

Main Research Question: How can Large Language Models be effectively applied to extract SPO triples from historical texts?

Large Language Models can be effectively applied to extract SPO triples from historical texts when their use is adapted to the complexities of historical language and structure. Effectiveness depends on the framework's design, its ability to handle nuanced context, and the alignment between the model's output and evaluation strategy.

Among the tested frameworks, **GPT o3 (via prompt engineering)** demonstrated the best balance of semantic accuracy and coverage, albeit limited by string-based evaluation. In contrast, **AiKG** over-generated and lacked precision, while **Triplex**, constrained by rigid prompts, under-generated critical relations. Success, therefore, hinges on flexible yet semantically robust models, paired with prompt strategies and evaluation methods that respect the historical domain's linguistic subtleties.

8.2 Takeaways

This project has demonstrated both the potential and complexity of applying Large Language Models to structured information extraction from historical texts. From a practical perspective, the process of identifying, testing, and evaluating LLM-based frameworks revealed that no system operates perfectly out of the box, each required careful configuration, adaptation, or post-processing to function effectively in the historical domain.

Working with historical texts posed unique challenges. The narrative density, varied syntax, and frequent use of indirect or archaic expressions made it difficult for many frameworks to detect relations that a human reader would easily infer. This emphasized the importance of not only selecting capable tools but also of building support structures around them, such as prompt design, data cleaning, and hybrid evaluation methods.

Another major takeaway is the importance of balancing automation with manual inspection. While metrics enabled consistent benchmarking, much of the insight came from directly reviewing extracted triples. This helped distinguish between actual model errors and limitations imposed by the evaluation script or the input formatting.

Overall, this research reinforced that LLMs are powerful but context-sensitive tools. Their effectiveness depends as much on thoughtful application as on raw model capability. With better alignment between extraction goals, prompt design, and evaluation strategy, LLMs can play a major role in the future of automated historical knowledge construction.

9. References

References

[1] Beatrice Heuser. The past as guide to the future. In *The Oxford Handbook of Grand Strategy*, pages 265–286. Oxford University Press, 2024.

[2] R. Ramachandran, K. Abhijith, and J. Karthik. Knowledge extraction from distributed heterogeneous data sources. In *2024 IEEE International Conference on Emerging Trends in Engineering and Technology (INCET)*, pages 1–6, 2024.

[3] M. Parnian and M. Z. Reformat. Triple extraction with generative technique for constructing weighted knowledge graph. In *2023 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, pages 127–134, 2023.

[4] Yifan Zeng. Histolens: An llm-powered framework for multi-layered analysis of historical texts – a case application of yantie lun. *arXiv preprint arXiv:2411.09978*, 2024.

[5] Issam El Naqa and Martin J. Murphy. What is machine learning? In *Machine Learning in Radiation Oncology: Theory and Applications*, pages 3–11. Springer International Publishing, 2015.

[6] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022.

[7] Shaoxiong Ji, Shirui Pan, Erik Cambria, Pasi Marttinen, and Philip S Yu. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems*, 33(2):494–514, 2022.
[8] Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, pages 1003–1011, 2009.
[9] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutiérrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Axel-Cyrille Ngonga Ngomo, Axel Polleres, Eric Prud'hommeaux, Juan F Sequeda, Steffen Staab, and Antoine Zimmermann. Knowledge graphs. *ACM Computing Surveys (CSUR)*, 54(4):1–37, 2021.
[10] OpenAI. Chatgpt o3 model. https://openai.com/chatgpt, 2024. https://openai.com/chatgpt.
[11] SandroGT. Kg-llm-prompting. https://github.com/SandroGT/KG-LLM-Prompting, 2023. https://github.com/SandroGT/KG-LLM-Prompting.
[12] Monarch Initiative. Ontogpt. https://github.com/monarch-initiative/ontogpt, 2023. https://github.com/monarch-initiative/ontogpt.
[13] SciPhi on HuggingFace. Triplex: Phi-3.8b-based triple extractor. https://huggingface.co/SciPhi/Triplex, 2024. https://huggingface.co/SciPhi/Triplex.
[14] victorlxh. Ickg v3.2. https://huggingface.co/victorlxh/ICKG-v3.2, 2024. https://huggingface.co/victorlxh/ICKG-v3.2.
[15] Clear-NUS. Edc: Extract–define–canonicalize. https://github.com/clear-nus/edc, 2023. https://github.com/clear-nus/edc.
[16] syrom. Localknowledgegraphextraction. https://github.com/syrom/LocalKnowledgeGraphExtraction, 2023. https://github.com/syrom/LocalKnowledgeGraphExtraction.
[17] ZJUNLP. Deepke. https://github.com/zjunlp/DeepKE, 2023. https://github.com/zjunlp/DeepKE.
[18] Robert McDermott. Ai powered knowledge graph generator. https://github.com/robert-mcdermott/ai-knowledge-graph, 2023. https://github.com/robert-mcdermott/ai-knowledge-graph.
[19] STAIR Lab. kg-gen: Knowledge graph generation from any text. https://github.com/stair-lab/kg-gen, 2023. https://github.com/stair-lab/kg-gen.
[20] Neo4j Labs. Knowledge graph builder. https://github.com/neo4j-labs/llm-graph-builder, 2024. https://github.com/neo4j-labs/llm-graph-builder.
[21] Ananya IIT Bhilai. Kgviz. https://github.com/Ananyaiitbhilai/KGViz, 2023. https://github.com/Ananyaiitbhilai/KGViz.
[22] Dylan Hogg. llmgraph. https://github.com/dylanhogg/llmgraph, 2023. https://github.com/dylanhogg/llmgraph.
[23] Edward Gibbon. *The History of the Decline and Fall of the Roman Empire*. Project Gutenberg, 1776.
[24] William Doyle. *The Oxford History of the French Revolution*. Oxford University Press, 1989.
[25] Anthony Beevor. *The Second World War*. Prussia Online, 2012.

Appendix

A AI Statement

Parts of this thesis were developed with the assistance of artificial intelligence tools, including OpenAI's ChatGPT (GPT-4-turbo and GPT-4o models). These tools were used primarily for language refinement, paragraph structuring, and the organization of findings in a coherent academic format.

All conceptual decisions, experimental design, framework implementation, manual evaluations, and final analysis were performed by the author. The use of AI was strictly limited to augmenting productivity and enhancing the clarity of expression. No parts of the thesis were generated without critical oversight and editing by the author, and all content has been verified for factual accuracy.

B Text Excerpts

Here you can find the text excerpts that the frameworks were run on.

B.1 The History of the Decline and Fall of the Roman Empire

We have already traced the emigration of the goths from Scandinavia, or at least from Prussia, to the mouth of the borysthenes, and have followed their victorious arms from the borysthenes to the danube. Under the reigns of valerian and gallienus, the frontier of the last-mentioned river was perpetually infested by the inroads of germans and sarmatians; but it was defended by the romans with more than usual firmness and success. The provinces that were the seat of war, recruited the armies of Rome with an inexhaustible supply of hardy soldiers; and more than one of these illyrian peasants attained the station, and displayed the abilities, of a general. Though flying parties of the barbarians, who incessantly hovered on the banks of the Danube, penetrated sometimes to the confines of Italy and Macedonia, their progress was commonly checked, or their return intercepted, by the imperial lieutenants. But the great stream of the gothic hostilities was diverted into a very different channel. The goths, in their new settlement of the ukraine, soon became masters of the northern coast of the euxine: to the south of that inland sea were situated the soft and wealthy provinces of asia minor, which possessed all that could attract, and nothing that could resist, a barbarian conqueror. The banks of the borysthenes are only sixty miles distant from the narrow entrance of the peninsula of crim tartary, known to the ancients under the name of chersonesus taurica. On that inhospitable shore, euripides, embellishing with exquisite art the tales of antiquity, has placed the scene of one of his most affecting tragedies. The bloody sacrifices of diana, the arrival of orestes and pylades, and the triumph of virtue and religion over savage fierceness, serve to represent an historical truth, that the tauri, the original inhabitants of the peninsula, were, in some degree, reclaimed from their brutal manners by a gradual intercourse with the grecian colonies, which settled along the maritime coast. The little kingdom of Bosphorus, whose capital was situated on the straits, through which the mæotis communicates itself to the euxine, was composed of degenerate Greeks and half-civilized barbarians. It subsisted, as an independent state, from the time of the Peloponnesian war, was at last swallowed up by the ambition of mithridates, and, with the rest of his dominions, sunk under the weight of the roman arms. From the reign of Augustus, the kings of Bosphorus were the humble, but not useless, allies of the empire. By presents, by arms, and by a slight fortification drawn across the isthmus, they effectually guarded, against the roving plunderers of sarmatia, the access of a country which, from its peculiar situation and convenient harbors, commanded the euxine sea and Asia minor. As long as the sceptre was possessed by a lineal succession of kings, they acquitted themselves of their important charge with vigilance and success. Domestic factions, and the fears, or private interest, of obscure usurpers, who seized on the vacant throne, admitted the goths into the heart of Bosphorus.

B.2 The Oxford History of The French Revolution

Although many provincial fédérés had taken part in the storming of the tuileries, the fall of the french monarchy had very largely been the work of the insurrectionary commune of paris. The very idea of a national convention to give france a republican constitution also

originated in the paris sections. It was therefore understandable that the sansculottes should regard themselves as the guardians and watchdogs of the new republic, and the arbiters of what it should stand for. And of course they were very well placed to enforce their will. The convention sat in paris and had no forces to defend itself from popular pressure. All available troops in 1792 and 1793 were occupied at the front, and the paris national guard was no longer the force that had shot down republican petitioners on the champ de mars. Since the end of july it had been open to all citizens and was little more than a sansculotte militia, commanded from 10 august by santerre, a rich brewer but long a popular activist in the city's east end. The legislative assembly had been forced to recognize its own helplessness in the face of parisian power during its last weeks. Its only attempt to assert itself, the decree dissolving the commune and ordering new elections on 30 august, was ostentatiously ignored and rapidly rescinded. And the deputies had had to sit powerless while the same sansculottes who claimed to be the nation's conscience massacred half the capital's prison population during the following week. The nation's representatives seemed to be in the clutches of a capricious and bloodthirsty mob, and in this respect the convention was no more secure than its predecessor. 'never forget', the ex-monk chabot warned his fellow deputies, 'that you were sent here by the sansculottes.' none of them was likely to; but they were deeply divided over whether that committed them to continue to do paris's bidding. The role of the capital in national affairs was to be the most persistently debated issue during the first nine months of the convention's existence. Leading the attack on paris were those who had sought to avert the insurrection of 10 august, and whom robespierre had tried to have arrested by the commune just as the prison massacres were beginning—men like brissot, vergniaud, and the 'faction of the gironde'. They had been deputies in the previous assembly, but they were supported by a number of newcomers, too. They were not a party, and never would be, except in the wishful imagination of their opponents; but they all sat for provincial constituencies, and the more prominent among them had grown used to informal co-operation with each other throughout the legislative. They tended to meet, as they had then, at the house of roland, still minister of the interior. There his pretty and ambitious wife, though a parisienne herself, railed constantly against marat, danton, robespierre, and the whole parisian delegation in the convention. These men, the girondins were convinced, had been deeply implicated in the september massacres, and intended to use their parisian support to seize national power.

B.3 The Second World War

The German 12th Infantry Division in front of Orsha pulled back just in time. When a major asked a pioneer officer why he was in such a hurry to blow a bridge after his battalion had crossed, the man handed him his binoculars and pointed across the river. Turning round, the major spotted a column of T-34 tanks, already within range. Orsha and Mogilev on the Dnepr were both cut off and taken in three days. Several hundred wounded had to be left behind. The German general ordered to hold Mogilev to the end was close to a nervous breakdown. Behind Soviet lines, the greatest problem was presented by the huge traffic jams of military vehicles. A broken-down tank could not be circumvented easily because of the marshes and forests either side of the roads. The chaos at times was such that 'the traffic controller at a crossroads might be a full colonel', a Red Army officer later recalled. He also pointed out how fortunate the Soviet forces were that there was so little sign of the Luftwaffe, since all those vehicles stuck nose to tail would have provided an easy target. On the southern flank, Marshal Rokossovsky's 1st Belorussian Front launched its assault with a massive preliminary bombardment which began at 04.00 hours. Explosions sent up fountains of earth. The ground was cratered and ploughed over a huge area. Trees came crashing down and German soldiers, instinctively adopting the foetal position in their bunkers, quivered as the ground vibrated as in an earthquake. Rokossovsky's northern pincer broke through between Tippelskirch's Fourth Army and the Ninth Army responsible for the Bobruisk sector. General der Infanterie Hans Jordan, the commander of the Ninth Army, brought in his reserve, the 20th Panzer Division. But as the counter-attack began that night, 20th Panzer was ordered to pull back and move south of Bobruisk. The penetration of the other pincer led by the 1st Guards Tank Corps had proved to be far more dangerous. It threatened to encircle the town and cut off the left flank of Ninth Army as well. Rokossovsky's surprise approach, through the edge of the Pripet Marshes, had a success similar to that of the Germans emerging from the Ardennes in 1940. Hitler still refused to allow retreat, so on 26 June Generalfeldmarschall Busch flew to Berchtesgaden to report to him at the Berghof. He was accompanied by Jordan, whom Hitler wanted to interrogate on his use of the 20th Panzer Division. But, while they were away from their headquarters, almost all of the Ninth Army was surrounded. The next day, both Busch and Jordan were dismissed. Hitler immediately resorted to the General-feldmarschall Model. Yet, even with this disaster and the threat to Minsk, the OKW still had no inkling of the scale of Soviet ambitions. Model, one of the few generals able to stand up to Hitler with success, was able to make the necessary withdrawals to the line of the River Berezina in front of Minsk.

C Ground Truth/Gold-standard Triples

In this section you can find the ground truth triples that helped in computing the evaluation metrics.

C.1 Roman Empire Ground Truth Triples

```
(Goths, emigrated from, Scandinavia)
(Goths, emigrated from, Prussia)
(Goths, moved from, mouth of Borysthenes)
(Goths, moved to, Danube)
(Valerian, ruled, the Roman Empire)
(Gallienus, ruled, the Roman Empire)
(Frontier of Danube, was infested by, Germans)
(Frontier of Danube, was infested by, Sarmatians)
(Frontier, was defended by, Romans)
(Provinces, recruited, Armies of Rome)
(Armies of Rome, supplied, Hardy soldiers)
(Illyrian peasants, attained, Station of general)
(Illyrian peasants, displayed, Abilities of general)
(Barbarians, hovered on, Banks of Danube)
(Barbarians, penetrated to, confines of Italy)
(Barbarians, penetrated to, confines of Macedonia)
(Imperial lieutenants, checked, Barbarians' progress)
(Imperial lieutenants, intercepted, Barbarians' return)
(Gothic hostilitites, were diverted, different channel)
(Goths, settled in, Ukraine)
(Goths, became masters of, Northern coast of the Euxine)
(Asia Minor, was located south of, Euxine)
(Asia Minor, possessed, Wealth)
```

(Asia Minor, lacked, Defense)
(banks of Borysthenes, was not distant from, Peninsla of Crim Tartary)
(Crim Tartary, was known as, Chersonesus Taurica)
(Euripides, placed, Tragedy scene on, Chersonesus Taurica)
(Diana, received, Bloody sacrifices)
(Orestes, arrived at, Chersonesus Taurica)
(Pylades, arrived at, Chersonesus Taurica)
(Virtue and religion, triumphed over, Savage fierceness)
(Tauri, were, original inhabitants of Chersonesus Taurica)
(Tauri, were reclaimed by, gradual intercourse with Grecian colonies)
(Grecian colonies, settled along, Maritime coast)
(Bosphorus, had capital on, Straits to the Euxine)
(Bosphorus, was composed of, Degenerate Greeks)
(Bosphorus, was composed of, Half-civilized barbarians)
(Bosphorus, was independent since, Peloponnesian War)
(Mithridates, conquered, Bosphorus)
(Bosphorus, lost to, Roman arms)
(Augustus, reigned, the Roman Empire)
(Kings of Bosphorus, were allies of, Roman Empire)
(Kings of Bosphorus, guarded against, plunderers of Sarmatia)
(Sarmatia, had, convenient harbors)
(Sarmatia, commanded, Euxine Sea)
(Sarmatia, commanded, Asia Minor)
(Kings, ruled with, vigilance and success)
(Domestic Factions, admitted, Goths)
(Usurpers, admitted, Goths)
(Goths, entered, Heart of Bosphorus)

C.2 French Revolution Ground Truth Triples

(provincl fédérés, took part in, storming of the Tuileries)
(fall of the French monarchy, was work of, insurrectionary Commune of Paris)
(national convention, gave, France a republican constitution)
(idea of a republican constitution for france, originated in, Paris sections)
(sansculottes, regarded themselves as, guardians of the new republic)
(sansculottes, were, arbiters of what the republic should stand for)
(sansculottes, were, well placed to enforce their will)
(convention, sat in, Paris)
(convention, lacked, forces to defend itself from popular pressure)
(available troops, were occupied at, the front in 1792)
(available troops, were occupied at, the front in 1793)
(Paris national guard, shot down, republican petitioners)
(Paris national guard, was open to, recruit any citizen)
(Paris national guard, was, little more than sansculotte militia)
(Paris national guard, was commanded by, Santerre)
(Santerre, was, rich brewer)
(Santerre, was, popular activist)
(legislative assembly, recognized, its helplessness)
(decree dissolving the commune, was, ignored)
(decree dissolving the commune, was, rescinded)
(sansculottes, massacred, half of the capital's prison population)
(deputies, were powerless against, sansculottes)
(Chabot, was, ex-monk)
(Chabot, warned, deputies)
(Chabot, stated, you were sent here by the sansculottes)
(deputies, were divided over, obedience to sansculottes)
(role of the capital, was, most debated issue in convention)
(Brissot, sought to avert, insurrection of 10 August)
(Vergniaud, sought to avert, insurrection of 10 August)
(Faction of the Gironde, sought to avert, insurrection of 10 August)
(Brissot, led, attack on Paris)
(Vergniaud, led, attack on Paris)
(Faction of the Gironde, led, attack on Paris)
(Robespierre, tried to have arrested, Brissot)
(Robespierre, tried to have arrested, Vergniaud)
(Robespierre, tried to have arrested, Faction of the Gironde)
(Robespierre, acted, as prison massacres were beginning)
(Brissot, were, former deputies)
(Vergniaud, were, former deputies)
(Faction of the Gironde, were, former deputies)
(Brissot, was supported, newcomers)
(Vergniaud, was supported, newcomers)
(Faction of the Gironde, was supported, newcomers)
(Brissot, sat for, provincial constituencies)
(Vergniaud, sat for, provincial constituencies)
(faction of the Gironde, sat for, provincial constituencies)
(Roland, was, minister of the interior)
(Girondins, met at, house of Roland)
(Roland's wife, was, pretty)
(Roland's wife, was, ambitious)
(Roland's wife, was, Parisienne)
(Roland's wife, railed against, Marat)
(Roland's wife, railed against, Danton)
(Roland's wife, railed against, Robespierre)
(Roland's wife, railed against, Parisian delegation)
(Girondins, believed, Parisian delegation was implicated in September massacres)
(Girondins, believed, Parisian delegation intended to seize national power)

C.3 WWII Ground Truth Triples

(German 12th Infantry Division, pulled back from, Orsha)
(Pioneer officer, was in a hurry to blow, bridge)
(Major, spoke to, Pioneer officer)
(Pioneer officer, handed, binoculars to Major)
(Pioneer officer, pointed across, river)
(Major, spotted, column of T-34 tanks)
(T-34 tanks, were, within range)
(Orsha, was cut off, three days)
(Orsha, taken in, three days)
(Mogilev, was cut off, three days)
(Mogilev, taken in, three days)
(Several hundred wounded, had to be, left behind)
(German general, was ordered to, hold Mogilev to the end)
(German general, was, close to a nervous breakdown)
(Greatest problem behind Soviet lines, was, traffic jams of military vehicles)
(Broken-down tank, could not be circumvented because of, marshes)
(Broken-down tank, could not be circumvented because of, forests)
(Red Army officer, recalled, chaos at crossroads)
(Traffic controller, might be, full colonel)
(Soviet forces, were fortunate due to, little sign of Luftwaffe)
(Vehicles, were, stuck nose to tail)
(Marshal Rokossovsky's 1st Belorussian Front, launched, assault)
(Preliminary bombardment, began at, 04.00 hours)
(Explosions, sent up, fountains of earth)
(Ground, was cratered, over a huge area)
(Ground, ploughed, over a huge area)
(Trees, crashed, down)
(German soldiers, adopted, foetal position)
(German soldiers, quivered as, ground vibrated)
(Rokossovsky's northern pincer, broke through, between Fourth Army)
(Rokossovsky's northern pincer, broke through, Ninth Army)
(General Hans Jordan, brought in, 20th Panzer Division)
(20th Panzer Division, was ordered to, pull back)
(20th Panzer Division, was ordered to, move south of Bobruisk)
(Penetration by 1st Guards Tank Corps, threatened to, encircle Bobruisk)
(Penetration, threatened to, cut off left flank of Ninth Army)
(Rokossovsky's approach, was through, edge of Pripet Marshes)
(Rokossovsky's success, was similar to, Germans in Ardennes in 1940)
(Hitler, refused to allow, retreat)
(Generalfeldmarschall Busch, flew to, Berghof)
(Busch, was accompanied by, Jordan)
(Hitler, wanted to interrogate, Jordan)
(Almost all of Ninth Army, was, surrounded)
(Busch, was, dismissed)
(Jordan, was, dismissed)
(Hitler, resorted to, Generalfeldmarschall Model)
(Model, made, withdrawals to River Berezina)
(OKW, had no inkling of, scale of Soviet ambitions)
(Model, was able to, stand up to Hitler with success)
(German forces, retreated from, Orsha)
(Orsha, was lost in, three days)
(Mogilev, was lost in, three days)
(Wounded soldiers, were left in, Mogilev)
(Soviet bombardment, began at, 04.00 hours)
(Explosions, caused, ground vibrations)
(Marshal Rokossovsky, commanded, 1st Belorussian Front)
(1st Belorussian Front, advanced through, Pripet Marshes)
(Soviet forces, achieved, breakthrough between Fourth Army and Ninth Army)
(Penetration, led to, encirclement of Bobruisk)
(Penetration, led to, cutting off of Ninth Army's left flank)
(Model, executed, strategic withdrawals)
(Model, resisted, Hitler's direct orders)
(German command, underestimated, Soviet offensive)
(Berghof, was residence of, Hitler)
(Hitler, dismissed, Busch and Jordan)
(German leadership, was in crisis after, losses on Eastern Front)
(Jordan, was interrogated by, Hitler)
(Soviet forces, faced, no Luftwaffe opposition)
(Military traffic jams, caused, operational chaos)
(Red Army logistics, suffered from, congestion)
(Soviet command structure, was disrupted by, chaotic vehicle flow)

D Results - Trimmed Examples

In this section you can find examples of the outputs that helped lead the behavioural analysis of the frameworks.

```
True Positives: 25
Predicted Triples: 39
Gold Triples: 71
Precision: 0.6410
Recall: 0.3521
F1 Score: 0.4545
Omission Rate: 0.6479
```

Fig. 4. Evaluation Metrics Computed for WW2 excerpt

```
Matched triples (predicted vs. gold):
['(Ground, was cratered, over a huge area)', '(Ground, was cratered, over a huge area)']
['(Explosions, sent up, fountains of earth)', '(Explosions, sent up, fountains of earth)']
```

Fig. 5. Example of GPT o3's Matched Triples

```
Possible matches (lenient threshold):
['(Hitler, wanted to interrogate, Hans Jordan on use of 20th Panzer Division)', '(Hitler, wanted to interrogate, Jordan)']
['(20th Panzer Division, was, reserve of Ninth Army)', '(20th Panzer Division, was ordered to, move south of Bobruisk)']
['(20th Panzer Division, was, reserve of Ninth Army)', '(Berghof, was residence of, Hitler)']
```

Fig. 6. Example of GPT o3's Possible Matchess

As you can see here, the possible matches are very leniently assigned, this aspect was mostly used to confirm that the triples were properly extracted, but the algorithm failed to match them. If they are in the Possible Matches list, then they are not recognized as full matches.

```
Possible matches (lenient threshold):
['(military vehicles, operate with, soldiers)', '(Military traffic jams, caused, operational chaos)']
['(german general, ordered, mogilev)', '(German general, was ordered to, hold Mogilev to the end)']
['(german general, ordered, mogilev)', '(German soldiers, quivered as, ground vibrated)']
['(german general, ordered, mogilev)', '(German command, underestimated, Soviet offensive)']
['(busch, dismissed via hitler, interrogate)', '(Busch, was, dismissed)']
['(rokossovsky's, emerged from, marshes)', '(Rokossovsky's approach, was through, edge of Pripet Marshes)']
['(rokossovsky's, emerged from, marshes)', '(German forces, retreated from, Orsha)']
['(retreat, dismissed soldiers, soldiers)', '(Hitler, dismissed, Busch and Jordan)']
['(jordan, commanded via army, luftwaffe)', '(Jordan, was interrogated by, Hitler)']
['(jordan, commanded via army, town)', '(Jordan, was interrogated by, Hitler)']
['(jordan, commanded via army, town)', '(Trees, crashed, down)']
['(jordan, commanded via army, town)', '(German command, underestimated, Soviet offensive)']
```

Fig. 7. Example of AiKG's Possible Matches

This example show why although hundreds of triples were extracted by AiKG, only a very small amount is considered a perfect match with the gold-standard triples.

E Prompt Used For GPT o3 Model

Below you will find the prompt used for triple SPO extraction via GPT o3 Model:

```
### SYSTEM ###
You are an expert information-extraction assistant.
Your only job is to read a passage and return every explicit or
unambiguous fact as knowledge-graph triples **in the form**
(Subject, Predicate, Object).

### THINKING STEPS  –  do NOT reveal
1. Read the passage once, then work sentence-by-sentence.
2. For each clause:
     Identify the grammatical subject and the main verb phrase.
     Identify the direct object, complement, or prepositional object.
3. Canonicalise:
   Use the most specific surface form for entities: (“United States Senate” > “Senate”).
4. Resolve coreference locally (pronouns, appositives, “the President”).
5. Include a triple only if **all three parts are explicit or can be
   resolved unambiguously from neighbouring text**.
6. Remove duplicates; preserve original order of appearance.

### OUTPUT FORMAT (exact) ###
(Subject, predicate, Object)
(Subject, predicate, Object)
. . .
• Use plain text, one triple per line.
• Do NOT output anything except triples – no bullet points, no JSON.

### START PASSAGE ###
{TEXT}
### END PASSAGE ###
### YOUR ANSWER ###
```