Consensus-Guided Knowledge Extraction from Aeronautical Accident Reports Using Heterogeneous Pretrained Language Models

Danilo de Farias Matos[1], Mauro Caetano[1], Daniel Alberto Pamplona[1]
[1]Instituto Tecnológico de Aeronáutica (ITA), São José dos Campos/SP – Brasil

Abstract - Aeronautical accident reports contain valuable safety knowledge, but their semi-structured language, domain-specific causal vocabulary, and heterogeneous report layouts make reliable automatic knowledge extraction difficult. This paper presents a training-free and auditable pipeline for extracting ontology-constrained knowledge triples from public reports issued by the Brazilian Aeronautical Accident Investigation and Prevention Center (CENIPA). Three heterogeneous pretrained language models independently produce candidate entities and relations from the same reports. Instead of modifying prompts after extraction, the proposed method applies post-hoc entity alignment, ontology validation, relation-label normalization, and a two-out-of-three consensus rule. In an offline reconstruction over 100 processed reports, relation-label normalization canonicalized 449 model outputs without any new LLM calls, increasing exact consensus triples from 192 to 220 and raising the exact consensus rate from 7.1% to 8.3%. The resulting graph remains intentionally conservative: 91.7% of ontology-valid candidates are retained as single-model audit records rather than inserted automatically. An additional semantic audit queue identifies 129 high-similarity candidate pairs for human adjudication. The contribution is therefore a reproducible post-processing methodology for transforming noisy pretrained LLM outputs into a smaller, traceable, and human-reviewable aviation safety knowledge graph.

Key Words - Knowledge Extraction, Large Language Models, Aviation Safety, Knowledge Graph, Consensus.

I. INTRODUCTION

Aeronautical accident investigation transforms evidence from past occurrences into knowledge intended to prevent recurrence. In Brazil, the Aeronautical Accident Investigation and Prevention Center [Centro de Investigação e Prevenção de Acidentes] (CENIPA, 2026) [1] publishes final reports that describe flight histories, analyses, conclusions, contributing factors, and safety recommendations. These reports form a valuable corpus for Natural Language Processing (NLP), but their technical vocabulary, historical variation in layout, and combination of structured fields with narrative explanations make automatic analysis difficult. Prior aviation NLP research has shown that safety reports can support classification, retrieval, and interactive analysis, while also requiring domain-aware processing strategies [2].

Large Language Models (LLMs) can perform relation extraction without task-specific parameter updates, reducing dependence on costly annotated corpora [3]. However, a single model may omit relevant relations, generate unsupported triples, or express equivalent entities and relations using different surface forms. SF-GPT addresses part of this problem through entity filtering, alias generation, and repeated-response self-fusion for knowledge graph construction [4]. This study adopts the same training-free paradigm, but shifts the central question from within-model self-fusion to cross-model agreement among heterogeneous pretrained models.

Matos. D, matosdfm@ita.br; Caetano. M, caetano@ita.br; Pamplona. D, pamplona@ita.br

That said, this paper proposes a consensus-guided strategy for extracting structured knowledge from CENIPA [1] reports. DeepSeek V4 Flash, GPT-4o mini, and OpenRouter Owl Alpha independently generate candidate triples under the same ontology-constrained extraction contract. Their outputs are then normalized, validated, and compared. Only triples supported by at least two models are admitted to the final graph, while unsupported candidates remain available for audit. The method deliberately avoids additional prompt tuning after the experiment; improvements are limited to deterministic post-processing of already-generated outputs.

Additionally, the contribution is deliberately domain-specific: multi-LLM consensus has recently been explored in clinical knowledge graph construction [5] and multi-agent graph enrichment [6], but its behavior remains insufficiently studied for training-free extraction from semi-structured aeronautical accident reports.

II. THEORETICAL BACKGROUND

A. NLP and Knowledge Extraction in Aviation Safety

Aviation related accident reports have long motivated NLP methods for classification, retrieval, and interactive safety analysis [2]. More recent studies apply LLMs to accident reasoning and causal identification. HFACS-guided Chain-of-Thought prompting improves the inference of human-factor patterns from general-aviation reports [7], while Retrieval-Augmented Generation improves causal identification from historical accident reports [8]. In parallel, transportation research has begun to frame LLM-assisted knowledge graph construction as a means of organizing fragmented and heterogeneous safety evidence, especially when reports contain incomplete or contradictory information [9]. Moreover, a recent aviation safety study further shows how topic modeling, LLM-based entity extraction, and knowledge graphs can be combined to uncover risks and policy gaps from investigation reports [10]. These approaches demonstrate the value of domain guidance and report preprocessing, but they do not address cross-model agreement in admitting triples into a knowledge graph.

Trusted extraction is especially relevant in aviation. A comparison of NLP and LLM tools for aircraft operations and maintenance identifies substantial limitations in zero-shot knowledge extraction and stresses the need for controlled, auditable methods [11]. Related accident-domain research shows that NLP can populate ontologies from heterogeneous narratives and support causal exploration [12]. Together, these studies motivate a pipeline that converts accident reports evidence into inspectable triples while explicitly controlling admission to the graph.

B. Training-Free Triple Extraction and Agreement

Zero-shot relation extraction treats an LLM as a general-purpose extractor controlled through instructions rather than

labeled training data. SumAsk method demonstrates that prompt reformulation can improve zero-shot relation extraction across model sizes and relation types [3]. SF-GPT method extends training-free extraction toward knowledge graph construction by filtering entities, generating aliases, and fusing repeated responses into self-fusion subgraphs [4]. More generally, text-to-knowledge graph generation has been modeled as a staged process in which nodes are identified before edge construction, reinforcing the importance of separating entity recognition, relation formation, and graph assembly decisions [13]. These mechanisms are central to the proposed method because equivalent surface forms must be aligned before agreement can be measured.

Agreement is commonly used to estimate annotation reliability, and LLM outputs can exhibit useful intercoder agreement in text-annotation tasks [14]. Yet agreement does not guarantee truth. Multiple models may share training data, implicit priors, or the same tendency to produce plausible causal statements. For this reason, the proposed pipeline treats agreement as a conservative admission rule, not as a substitute for evidence grounding or expert review.

C. Positioning Against Multi-LLM Knowledge Graph Methods

The closest recent line of studies combine multi-LLM consensus validation, ontology alignment, and retrieval-augmented generation for clinical knowledge graph construction [5], while the KARMA method similarly coordinates specialized LLM agents for entity discovery, relation extraction, schema alignment, and conflict resolution in knowledge graph enrichment [6]. On the other hand, applied transportation studies also show that LLMs can structure accident information into graph form, including road-accident knowledge graphs from news articles [15] and aviation-equipment fault-diagnosis graphs that integrate joint extraction with LLM support [16]. In contrast, the present study isolates a simpler and more reproducible question: can heterogeneous pretrained models, followed by deterministic normalization and two-out-of-three agreement, yield a reliable graph from semi-structured aviation accident reports?

This study's contribution is therefore narrower, but methodologically useful. It does not require retrieval, fine-tuning, or multi-agent orchestration. It measures the gains from using pretrained models as independent extractors and curating their outputs through ontology-aware post-processing.

III. METHOD

A. Data Source and Preprocessing

The corpus consists of public final reports issued by CENIPA [1]. Simplified reports are prioritized because they provide a relatively homogeneous subset and consistently expose sections relevant to knowledge extraction. PDF files are converted to a linear text representation suitable for sequence processing. Unicode is normalized, line-break hyphenation is repaired, and report sections such as flight history, analysis, conclusions, contributing factors, and safety recommendations are retained when available.

A stratified sample was selected to reduce evaluation bias across report content. Reports are represented using TF-IDF features, projected into a lower-dimensional space, and clustered. Samples are then drawn from the resulting groups. This stage is used only to obtain a diverse evaluation subset and does not train or fine-tune the extraction models.

This process produces 100 text files as raw input. Each document is submitted to the model ensemble once. The resulting model outputs are saved as JSON logs that include raw responses, parsed entities, parsed relations, evidence spans, source-section labels, entity-existing-filter (EEF) rejections, and request metadata. Subsequent analyses in this paper use an offline reconstruction of those saved outputs, so no additional model calls are needed to evaluate post-processing changes.

B. Candidate Triple Generation

Each report is submitted independently to DeepSeek V4 Flash, GPT-4o mini, and OpenRouter Owl Alpha through OpenRouter. All models receive the same ontology-constrained extraction contract, temperature setting, JSON output requirement, and relation inventory. No model receives another model's answer. Each candidate relation must contain a head entity, a relation label, a tail entity, supporting evidence, and a source-section identifier.

The initial extraction prompt follows the training-free logic of SF-GPT: the model first identifies relevant entities, then proposes relations among those entities, and finally fuses each triple in a subgraph that is compared among the other models' responses to verify if the output is explicitly supported by the guiding ontology.

It is necessary to note that the extraction stage is intentionally kept fixed for the final analysis. The goal is not to improve results by revising the prompt after observing model behavior. Instead, the study asks whether deterministic post-processing can transform pretrained model outputs into a more reliable and auditable graph.

C. Entity, Relation, and Ontology Normalization

Before voting, candidate triples were normalized to prevent superficial wording differences from appearing as disagreement. Entity aliases were aligned using canonical forms inspired by the alias-generation stage of SF-GPT [4]. Entity types were then normalized against the ontology class catalog, and each relation validated against ontology domain and range constraints.

The implementation applies normalization in a fixed order. First, the Entity Extraction Filter (EEF) retains a relation only when both its head and tail occur in that model's extracted entity list, following SF-GPT's use of the entity inventory as a check on generated triples [4]. The canonical label is stored together with the aliases returned for the entity. For two alias lists, every cross-list pair is compared with normalized Levenshtein similarity and the scores are aggregated. Entities are aligned when the aggregate score is at least $s = 0.55$. The alias-based similarity function follows SF-GPT [4], but 0.55 is the value stored in the experiment configuration; it was used in the offline reconstruction and was neither copied literally from SF-GPT's 0.4 cross-text threshold nor tuned on the 100 CENIPA reports. Instead, a higher threshold compared to the literature was chosen to further enforce alias alignment before considering an entity for inclusion. For example, an aircraft represented as 'PT-XYZ' with aliases 'aircraft' and 'the airplane' can be aligned with another output that preserves the registration and an overlapping aircraft alias, whereas an isolated generic word without supporting aliases is not sufficient by itself.

Second, entity types are stripped of surrounding whitespace and matched case-insensitively to the ontology class catalog, restoring the exact ontology spelling when a match exists. The relation must be an authorized object property, both endpoint types must be known, and self-loops or relations whose endpoints have the same semantic type are rejected. Domain and range are evaluated over the transitive class hierarchy, so a subclass is accepted when it inherits from an authorized domain or range. Thus, an Aircraft involvedIn an Occurrence is structurally valid, whereas Aircraft occurredAt Infrastructure is rejected because occurredAt requires an Occurrence as its head. These checks are categorical constraints, not score thresholds: a candidate either satisfies the ontology contract or is retained in the rejection log with its reason.

Third, relation labels are canonicalized deterministically. A relaxed label is lowercased for matching and stripped of punctuation and whitespace; it is mapped only when the resulting key corresponds to one authorized ontology property. Accordingly, 'involved in', 'involved-in', and involvedIn resolve to involvedIn; 'occurred at' resolves to occurredAt; and 'has consequence' resolves to hasConsequence. No endpoint, relation meaning, or evidence span is invented. If the normalized key is absent from the ontology inventory, the relation is rejected rather than approximated. This layer therefore corrects lexical variation while keeping factual admission separate from normalization.

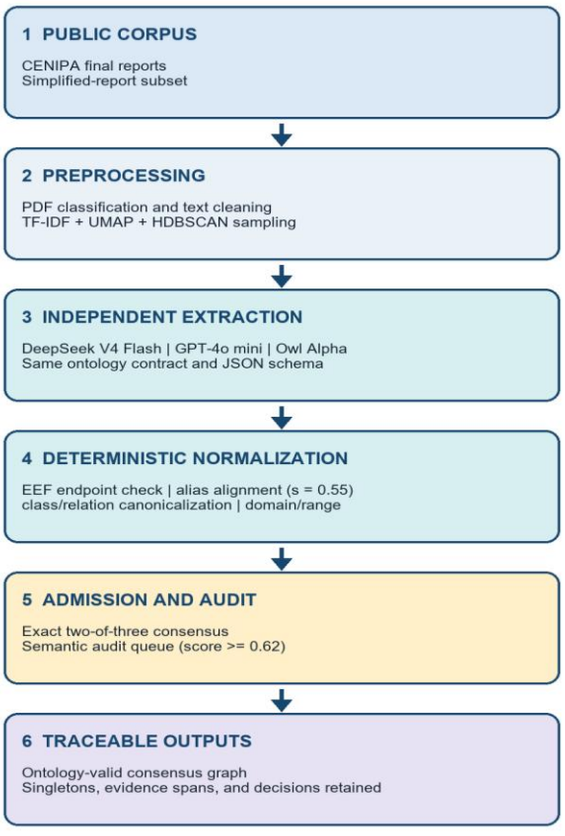

Fig. 1. End-to-end pipeline from public CENIPA reports to the consensus graph and supervised semantic audit queue.

Figure 1 makes explicit which stages generate candidate knowledge and which stages only curate saved outputs. Model calls occur only at the independent-extraction stage. EEF, alias alignment, class and relation canonicalization, ontology validation, exact voting, and semantic ranking are subsequently reproducible offline; the semantic queue never bypasses human review.

D. Two-of-Three Consensus and Semantic Audit

Let T_m be the normalized triple set produced by model m, and let M be the set of three models. For each candidate triple t in the union of all normalized outputs, support is defined as $support(t) = |\{m \; in \; M : t \; in \; T_m\}|$. The exact consensus graph admits t when $support(t) \geq 2$ and the triple satisfies ontology validation. Triples with $support(t) = 1$ remain in the audit log but are not inserted into the final graph. Triples with $support(t) = 3$ are marked as unanimous high-agreement outputs.

A second layer identifies candidates for semantic audit among single-model outputs. Candidate pairs are grouped by relation and by comparable ontology superclasses for head and tail entities. Within each group, TF-IDF character n-gram vectors are built for head labels, tail labels, aliases, and evidence text, then cosine similarities are combined with endpoint labels carrying 90% of the weight and evidence carrying 10%. The audit cutoff of 0.62 was used as an empiric operational prioritization rule to produce a reviewable high-similarity queue. Like the entity-alignment threshold s = 0.55 stored in the experiment configuration, it was used in the offline reconstruction and is not claimed as a literature-derived optimum or an accuracy guarantee. Pairs above 0.62 are never inserted automatically and remain subject to expert adjudication and future sensitivity analysis.

IV. RESULTS

A. Effect of Relation-Label Normalization

The reconstruction processed 100 reports and rebuilt all consensus artifacts from saved model outputs. Relation-label normalization canonicalized 449 relation labels, all from GPT-4o mini. The three most frequent mappings were ‘involved in’ to ‘involvedIn’ (151), ‘has contributing factor’ to ‘hasContributingFactor’ (138), and ‘occurred at’ to ‘occurredAt’ (74).

Before this normalization layer, the exact consensus graph contained 192 accepted triples. After normalization, it contains 220 accepted triples, an increase of 28 triples. The number of documents with at least one exact consensus triple increased from 49 to 53. Unanimous triples doubled from 8 to 16, indicating that a portion of apparent disagreement was caused by relation-label surface variation rather than substantive extraction disagreement. This effect can be better visualized in Figure 2.

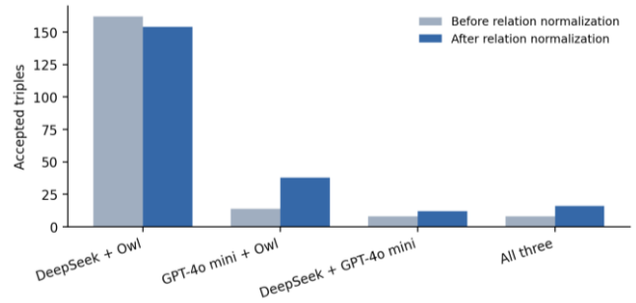

Fig. 2. Accepted triples by supporting model combination before and after deterministic relation-label normalization.

Figure 2 shows that relation-label normalization mainly reveals hidden agreement involving GPT-4o mini. The increase in unanimous and GPT-supported triples indicates that part of the initial disagreement was lexical rather than semantic.

B. Conservative Graph Admission

After normalization, the pipeline produced 2,677 candidate triples after entity-level extraction and parsing. Of these, 15 were rejected by ontology validation, leaving 2,662 ontology-valid candidates for consensus analysis. The exact consensus graph admitted 220 triples, corresponding to an exact consensus rate of 8.3%. The remaining 2,442 ontology-valid candidates were retained as single-model audit records rather than automatically inserted into the graph.

This result should be interpreted as a precision-oriented graph-construction strategy rather than a high-recall extractor. The method deliberately reduces automatic graph coverage so that admitted triples have cross-model support and ontology-compatible structure. The filtering rate of 91.7% represents claims that are not discarded from the experiment but withheld from the final graph until additional evidence or human review supports them, as shown in Figure 3.

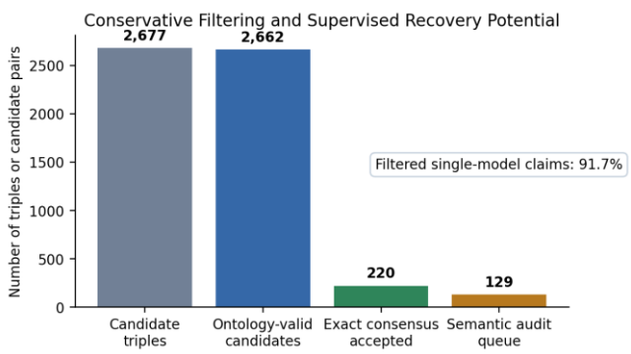

Fig. 3. Filtering funnel for the normalized evidence base, showing exact graph admission and the supervised semantic audit queue

Figure 3 highlights the conservative nature of the proposed graph-admission strategy. Most ontology-valid single-model claims are withheld from automatic insertion, which motivates the semantic audit layer as a supervised recovery mechanism.

C. Model-Level Behavior

The three models contributed differently to the final graph. DeepSeek V4 Flash produced valid outputs for 85 documents and failed for 15 due to JSON parsing errors. GPT-4o mini and Owl Alpha returned parseable outputs for all 100 documents. DeepSeek V4 Flash produced 982 raw relation records with 5 entity-existence-filter rejections; GPT-4o mini produced 880 raw relation records with 104 rejections and 449 relation-label normalizations; Owl Alpha produced 1,410 raw relation records with 35 rejections.

The normalization layer had its strongest effect on GPT-4o mini. Before normalization, GPT-4o mini often expressed ontology relations in natural-language forms. After canonicalization, the GPT-4o mini + Owl Alpha accepted-pair count increased from 14 to 38, and the DeepSeek + GPT-4o mini accepted-pair count increased from 8 to 12. This supports the methodological claim that post-processing can recover hidden agreement without changing prompts or generating new model outputs, and the models' behavior can be better visualized in Figure 4.

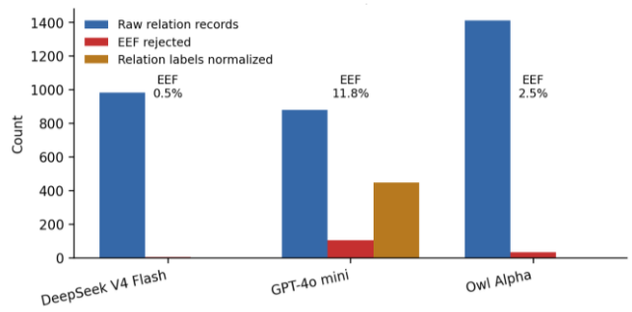

Fig. 4. Model-level relation records, entity-existence-filter rejections, and deterministic relation-label normalizations.

Figure 4 compares model behavior after reconstruction, including relation volume, entity-existence-filter rejections, and normalized relation labels. The results show why post-processing is especially important for harmonizing GPT-4o mini with the other models.

D. Semantic Audit Candidates

The normalized evidence base contains 129 semantic audit candidates across 46 documents. These candidates represent 58.6% of the exact consensus graph size, indicating a substantial supervised expansion opportunity if human reviewers approve them. Importantly, semantic candidates are not automatically inserted into the final graph; they instead define a prioritized audit queue for human adjudication.

For each pair of structurally comparable singleton triples, semantic similarity is computed from three cosine-similarity components: head-entity similarity, tail-entity similarity, and evidence-span similarity. Each text field is represented with TF-IDF character n-grams using a word-boundary analyzer and n-grams of length 3 to 5. Let sim_h be the cosine similarity between the head labels and aliases, sim_t the cosine similarity between the tail labels and aliases, and sim_e the cosine similarity between the supporting evidence spans. The final semantic audit score is defined as:

$$score = \left(\frac{1-w_e}{2}\right) sim_h + \left(\frac{1-w_e}{2}\right) sim_t + w_e . sim_e \qquad (1)$$

where w_e is the evidence weight. In this experiment, $w_e = 0.1$, so the head and tail similarities each contribute 45% of the final score, while evidence similarity contributes 10%. A candidate pair is retained in the semantic audit queue when $score \geq 0.62$ and when the two triples come from different model outputs without model-overlap support. Among the retained candidates, the mean score is 0.750 and the median is 0.730.

The strongest semantic candidates mostly involve boundary or alias differences rather than new relations. Examples include event labels such as engine failure versus loss of engine power, location labels with or without aerodrome codes, and consequence labels with different capitalization or specificity. Lower-scoring candidates are more ambiguous and should remain subject to manual adjudication. This reinforces the decision to keep semantic similarity as an audit queue rather than as an automatic graph-insertion rule.

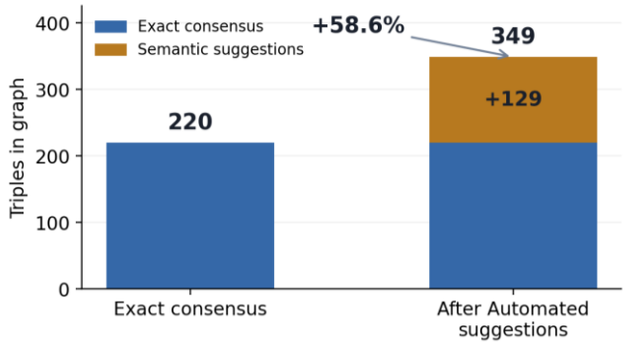

Fig. 5. Potential graph expansion after human validation.

Figure 5 illustrates the potential expansion of the graph if all semantic suggestions were accepted after review. It does not represent automatic graph insertion but rather the maximum supervised gain available from the semantic audit queue.

V. DISCUSSION

The results suggest that the main benefit of the proposed approach is not maximal extraction volume. The method produces a smaller graph, but one whose edges have explicit cross-model support, ontology-compatible structure, evidence provenance, and a traceable audit trail. In aviation safety analysis, this trade-off is attractive because unsupported or poorly grounded triples can distort causal interpretation.

The relation-label normalization experiment is particularly important for the paper's contribution. It shows that some disagreement among pretrained models is lexical rather than semantic. A deterministic canonicalization layer recovered hidden agreement from already generated outputs and increased exact consensus without changing the extraction prompt. This supports the central methodological position: pretrained models can produce more useful knowledge graphs when their outputs are normalized and adjudicated after generation.

At the same time, the low exact consensus rate reveals substantial model diversity and instability. Many candidate triples remain single-model outputs. Some may be valid omissions by other models; others may be unsupported inferences or overly broad entity boundaries. The semantic audit queue provides a practical way to focus human review on candidates that are structurally comparable and textually similar, instead of asking experts to inspect thousands of isolated singletons.

The comparison with existing methods is methodological rather than a head-to-head accuracy benchmark. Conventional supervised relation extractors can report precision and recall against annotated datasets but require relation-specific labels and may transfer poorly to a new domain [4], while SumAsk changes prompt formulation to improve zero-shot extraction [3]. SF-GPT instead repeats one model and fuses its responses using EEF and entity aliases [4]. The present method retains those filtering ideas but replaces within-model repetition with independent outputs from heterogeneous pretrained models and makes two-of-three agreement the automatic admission rule. Compared with retrieval-augmented multi-LLM construction [5] and multi-agent enrichment [6], it deliberately omits retrieval and agent orchestration so that every post-processing decision can be reconstructed from saved outputs. Because datasets, ontologies, and relation inventories differ, these contrasts establish scope and design trade-offs, not statistical superiority.

Operationally, the graph converts narrative reports into traversable associations among occurrences, aircraft, locations, contributing factors, consequences, and safety recommendations. An investigator examining runway excursion can query the runway-excursion node for adjacent and two-hop nodes, rank recurrent contributing factors, and retrieve the reports and evidence spans supporting each edge. The same structure can support semantic retrieval of cases whose wording differs, reveal repeated factor-occurrence patterns, and connect a candidate hazard to prior recommendations. These functions complement, rather than replace, documentary analysis: provenance and the audit status of each edge remain visible so an investigator can return to the source passage before using a relationship in an operational conclusion.

The present experiment uses public CENIPA final reports. Applying the pipeline to confidential investigation material or personal data requires a separate governance design: minimization and de-identification before model submission, documented legal authority and purpose limitation, access-controlled logs, retention and deletion rules, and verification of each provider's data-use and privacy policies. When external processing is not authorized, the extraction stage should run in an approved private or local environment. Consensus among providers does not mitigate disclosure risk, so privacy review must precede model selection and data transfer.

VI. LIMITATIONS AND NEXT STEPS

This study does not report precision, recall, false-positive rate, or false-negative rate because no expert-annotated gold standard exists for the CENIPA relation inventory in the current experiment. The reported counts, ontology-rejection rate, filtering rate, and cross-model agreement are internal pipeline measures; they quantify processing behavior, not extraction correctness. In particular, the 91.7% withholding rate must not be interpreted as a hallucination or false-positive rate, and disagreement cannot be counted as a false negative without an independently annotated reference set.

A dedicated follow-up study is being prepared for future publication with qualitative evaluation by aviation safety specialists. Its design follows the relation-extraction evaluation logic of Trusted knowledge extraction for operations and maintenance intelligence [11], which used domain-expert manual assessment when a universal relation gold standard was unavailable. Experts will review a stratified sample of exact-consensus triples, singletons, semantic-audit candidates, and ontology rejections, while blinded to the originating model combination. Each item will be assessed for syntactic validity of the relation endpoints, semantic accuracy against the report and domain knowledge, consistency with the other triples from the same report, and sufficiency of the cited evidence span. Adjudicated labels will subsequently permit precision estimates for each stratum, approval yield for semantic candidates, and an analysis of disagreements among experts.

For reproducibility, a public repository was released (https://github.com/matosdf/SIGE-MMA-V2) and the complete source snapshot used in this study was pinned under the V.2 release tag. It includes the preprocessing and post-

processing scripts, the ontology and configuration files, and a minimal replication example based on public reports. The release also documents software dependencies, fixed random seed, clustering parameters, normalization thresholds, and expected intermediate JSON artifacts. API credentials and provider-controlled model services will not be distributed; their versions and request parameters will be recorded so that external dependencies remain explicit.

VII. CONCLUSION

This paper presents a training-free, consensus-guided method for extracting ontology-constrained knowledge triples from CENIPA accident reports. Three heterogeneous pretrained LLMs independently generate candidate triples, and deterministic post-processing aligns entities, canonicalizes relation labels, validates ontology structure, and admits only two-out-of-three agreements to the final graph.

In the normalized offline reconstruction over 100 reports, the method produced 220 exact consensus triples and 129 semantic audit candidates. Relation-label normalization recovered hidden agreement, especially involving GPT-4o mini, while preserving the conservative graph-admission rule. The resulting graph is way smaller than the union of model outputs, but more auditable and better suited to human-supervised safety knowledge construction.

The broader contribution is a methodology for using pretrained models without fine-tuning or post-hoc prompt optimization: let models disagree, normalize what can be normalized deterministically, admit only supported and ontology-valid triples, and convert the remaining near-agreements into a focused human review queue.

REFERENCES

[1] Aeronautics Command – Brazilian Air Force, "CENIPA's Final Reports," World Wide Web. Accessed: May 20, 2026. [Online]. Available: https://sistema.cenipa.fab.mil.br/cenipa/paginas/relatorios/relatorios.php>).

[2] L. Tanguy, N. Tulechki, A. Urieli, E. Hermann, and C. Raynal, "Natural language processing for aviation safety reports: From classification to interactive analysis," *Comput. Ind.*, vol. 78, pp. 80–95, May 2016, doi: 10.1016/j.compind.2015.09.005.

[3] G. Li, P. Wang, and W. Ke, "Revisiting Large Language Models as Zero-shot Relation Extractors," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds., Singapore: Association for Computational Linguistics, Dec. 2023, pp. 6877–6892. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.459/

[4] L. Sun, P. Zhang, F. Gao, Y. An, Z. Li, and Y. Zhao, "SF-GPT: A training-free method to enhance capabilities for knowledge graph construction in LLMs," *Neurocomputing*, vol. 613, Jan. 2025, doi: 10.1016/j.neucom.2024.128726.

[5] U. Das, K. B. Atmakuri, D. Ho, C. Lee, and Y. Lee, "Clinical Knowledge Graph Construction and Evaluation with Multi-LLMs via Retrieval-Augmented Generation," 2026. [Online]. Available: https://arxiv.org/abs/2601.01844

[6] Y. Lu, W. Wu, X. Zhao, R. Peng, and J. Wang, "KARMA: Leveraging Multi-Agent LLMs for Automated Knowledge Graph Enrichment," 2026. [Online]. Available: https://arxiv.org/abs/2502.06472

[7] Q. Liu, F. Li, K. K. H. Ng, J. Han, and S. Feng, "Accident investigation via LLMs reasoning: HFACS-guided Chain-of-Thoughts enhance general aviation safety," *Expert Syst. Appl.*, vol. 269, Apr. 2025, doi: 10.1016/j.eswa.2025.126422.

[8] T. Ren, Z. Zhang, B. Jia, and S. Zhang, "Retrieval-Augmented Generation-aided causal identification of aviation accidents: A large language model methodology," *Expert Syst. Appl.*, vol. 278, Jun. 2025, doi: 10.1016/j.eswa.2025.127306.

[9] Y. Ling, Z. Qin, and Z. Ma, "A review of knowledge graph construction using large language models in transportation: Problems, methods, and challenges," *Transp. Res. Part C Emerg. Technol.*, vol. 183, p. 105428, Feb. 2026, doi: 10.1016/j.trc.2025.105428.

[10] S. Farzadnia, R. Merkert, and M. J. Beck, "Analysing aviation safety reports for uncovering current risks and policy gaps using topic modelling and knowledge graphs," *Saf. Sci.*, vol. 200, p. 107230, Aug. 2026, doi: 10.1016/j.ssci.2026.107230.

[11] K. P. Mealey, J. A. Karr, P. Saboia Moreira, P. R. Brenner, and C. F. Vardeman, "Trusted knowledge extraction for operations and maintenance intelligence," *Natural Language Processing Journal*, vol. 13, Dec. 2025, doi: 10.1016/j.nlp.2025.100187.

[12] J. I. Single, J. Schmidt, and J. Denecke, "Knowledge acquisition from chemical accident databases using an ontology-based method and natural language processing," *Saf. Sci.*, vol. 129, Sep. 2020, doi: 10.1016/j.ssci.2020.104747.

[13] I. Melnyk, P. Dognin, and P. Das, "Knowledge Graph Generation From Text," in *Findings of the Association for Computational Linguistics: EMNLP 2022*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds., Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, Dec. 2022, pp. 1610–1622. doi: 10.18653/v1/2022.findings-emnlp.116.

[14] F. Gilardi, M. Alizadeh, and M. Kubli, "ChatGPT outperforms crowd workers for text-annotation tasks," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 120, no. 30, Jul. 2023, doi: 10.1073/pnas.2305016120.

[15] J. Yang and S. Park, "LLM-Assisted Knowledge Graph Construction for Managing Road Accident Information From News Articles," *IEEE Access*, vol. 13, pp. 217846–217865, 2025, doi: 10.1109/ACCESS.2025.3648842.

[16] L. Li, A. Haruna, W. Ying, K. Noman, and Y. Li, "Knowledge graph-driven fault diagnosis for aviation equipment: Integrating improved joint extraction with large language model," *J. Ind. Inf. Integr.*, vol. 50, Mar. 2026, doi: 10.1016/j.jii.2025.101039.