Towards Hybrid Ontology Extraction

Leveraging Human-AI Synergy for Knowledge Graph Construction

Amal Sayed Ebrahim Salman Maki 2839123
Ramya Ragu 2836480

Vrije Universiteit - Amsterdam
a.s.e.s.maki@student.vu.nl
r.ragu@student.vu.nl

Abstract. Ontology extraction from literature is a vital yet intricate task that necessitates an equilibrium in semantic depth and scalability. While manual methods provide high percision, they are labor-intensive and often are constrained by cognitive limitations such as bias, error proneness, and short memory. These challenges can hinder consistency and fairness in decision-making, highlighting the need for hybrid approaches where machine-driven inference complements human judgment with greater efficiency and generalizability [22]. On the other hand, fully automated systems, particularly those utilizing Large Language Models (LLMs), face challenges with semantic fidelity and domain coverage. The hybrid approach utilizes human expertise to define domain-specific structures while relying on LLMs for automated Named Entity Recognition (NER) and the extraction of semantic relationships, which are further refined using similarity metrics and confidence scoring. Experimental evaluations compare manual, automated, and hybrid models across essential Natural Language Processing (NLP) tasks, demonstrating that the hybrid approach significantly minimizes inconsistencies, improves semantic quality, and scales more effectively across extensive corpora. This proposed pipeline offers a promising approach to generating scalable, high-quality ontologies applicable across various research fields.

Keywords: Large Language Models · Named Entity Recognition · Semantic Relationship Extraction · Ontology · Natural Language Processing

1 Introduction

Hybrid Intelligence (HI) refers to the synergistic collaboration between humans and machines to achieve goals neither could reach alone [8]. As AI systems become more deeply integrated into everyday human activities, supporting reasoning, decision-making, and task execution, the need to structure and represent knowledge flow within human-AI teams (hybrid intelligence) becomes increasingly critical. Ontologies and knowledge graphs address this need by enabling formal, machine-readable reasoning while preserving human interpretability.

This project investigates a hybrid approach to ontology extraction that integrates the contextual awareness of human annotators with the scalability and speed of large language models (LLMs). As noted in the Human-Machine Intelligence (HMI) paradigm, knowledge engineering (KE) must evolve to support symbolic, subsymbolic, and human-in-the-loop components. However, despite its potential, hybrid intelligence lacks a shared methodology for engineering knowledge exchange between human-machine teams.

A major obstacle in ontology design is its traditionally manual nature: it is time-intensive, cognitively demanding, and requires deep domain expertise [7, 6]. Automated methods, while efficient, often lack contextual nuance or semantic rigor. This tension leads to our central research question:

> *How can human-AI collaboration be leveraged to improve the quality, efficiency, and semantic expressiveness of ontology extraction?*

To explore this, we applied a hybrid pipeline to a curated corpus of ten academic papers. These papers cover diverse perspectives on explainability, ethics, trust, legal reasoning, semantic perception, and human-machine collaboration. The extracted knowledge forms the foundation of a hybrid ontology designed to support reasoning in hybrid intelligence scenarios.

This report is organized as follows: Section 2 details the data and ontology design. Section 3 presents insights from SPARQL queries and metrics. Section 4 outlines our research proposal, and Section 5 concludes with limitations and future directions.

2 Data

To guide our hybrid ontology construction process, we analyzed and extracted knowledge from a curated corpus of ten academic papers, each addressing critical aspects of hybrid intelligence. These works span themes such as explainability, ethics, law, perception, education, and collaborative reasoning in AI systems. The following summaries highlight the core contributions of each document:

- **Ethical Preferences in the Digital World: The EXOSOUL Questionnaire** [4] – Introduces a semantic model and a questionnaire to capture individual ethical preferences, forming a bridge between personal values and AI behavior.
- **Exemplars and Counterexemplars Explanations for Skin Lesion Classifiers** [17] – Explores interpretable machine learning through the use of visual examples, highlighting the role of counterfactual reasoning for enhanced explainability.
- **Building a Stronger Case: Combining Evidence and Law in Scenario-Based Bayesian Networks** [23] – Presents a legal reasoning system combining probabilistic inference with normative legal knowledge to support structured decision-making.

- **The Games We Play: How Critical Complexity Improves Machine Learning** [9] – Discusses critical complexity theory and its implications for designing AI systems that adapt to rich, unpredictable environments.
- **Unreflected Acceptance—Investigating the Negative Consequences of ChatGPT-Assisted Problem Solving in Physics Education** [20] – Examines how reliance on generative AI without reflection can lead to surface-level learning, and poor academic performance.
- **Legal-Ethical Challenges and Technological Solutions to Health Data Consent in the EU** [10] – Highlights the need for transparent, law-compliant data governance, offering technological frameworks to manage consent and accountability.
- **A Conceptual Model for Implementing Explainable AI By Design** [24] – Proposes a framework for embedding explainability into the design phase of AI systems, emphasizing structured justification mechanisms and transparency.
- **Knowledge Graphs in Support of Human-Machine Intelligence** [8] – Explores how knowledge graphs enable mutual understanding between humans and machines, especially in decision support contexts.
- **Egocentric Hierarchical Visual Semantics** [12] – Proposes a model for semantic perception that links visual experience to symbolic representations — relevant for understanding hybrid reasoning from perception to ontology.
- **Open, Multiple, Adjunct. Decision Support at the Time of Relational AI** [5] – Investigates how relational AI systems can support complex, open-ended decision-making by combining formal logic with contextual reasoning.

Together, these documents form a diverse, yet interconnected foundation for designing and populating a hybrid ontology that captures foundational research topics in AI, such as ethical paradigms, reasoning structures, and explainability mechanisms in AI systems.

2.1 Ontology Design

The original ontology summary consisted of 15 classes, 13 properties, and 28 individuals, summing up to 114 triple counts. However, based on the newly acquired dataset containing 10 research papers, significant expansions and modifications were made to enhance the ontology's representation of domain knowledge.

After processing the information, the final ontology summary consists of 146 classes, 198 properties, and 473 individuals, forming a sum of 811 triples. The final knowledge graph structure is shown in Figure 1

Object properties in this model were designed using a triangular modeling pattern where applicable. That is, if R(ab) and R(ac) exist, then R(bc) is also explicitly modeled. Example if :AIUser :uses :ChatGPT, :ChatGPT :hasConsequence :Weakened-Learning-Outcomes, then the ontology also includes the derived relationship: :AIUser :experiences :Weakened-Learning-Outcomes. This

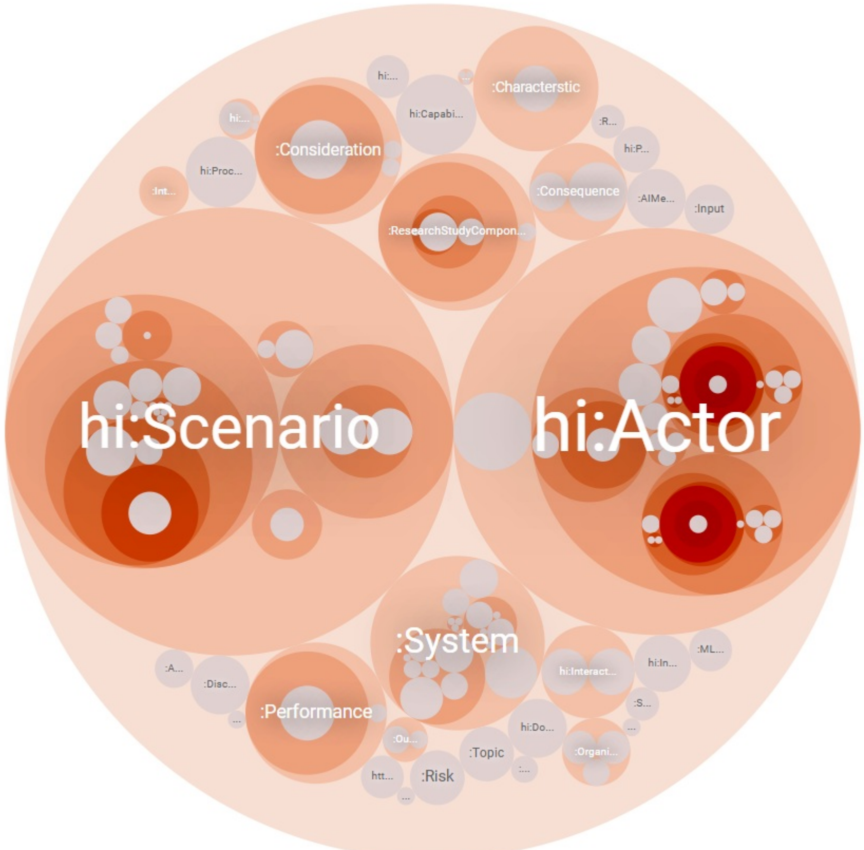


Fig. 1: Hybrid Intelligence Knowledge Graph

approach ensures that such indirect effects are not only inferable but also explicitly represented with meaningful labels to enhance the expressiveness of the ontology.

The process of ontology enhancement involved manual scraping to extract relevant concepts and relationships from the research papers. This is followed by automated annotations of individuals based on manually extracted concepts and relationships.

Classes :Human and :ArtificialAgent are two central classes in the ontology. Figure 2 shows the final design of these classes.

The final ontology is organized with improvisation of class structure and categorization. In the Figure 3 the following classes are prioritized:

- **Characteristic:** Represents attributes or properties that describe various entities within the ontology such as :Problem, :Human, :System or :ArtificialAgent.
- **Consequence:** Captures negative outcomes or effects experienced by a :User while interacting with a :System.
- **Risk:** In contrast with :Consequence, :Risk references the threats the :System poses to the system :User.
- **Scenario:** Represents different objectives of the system. It comprises new contextual setups, further categorized into:
  - **Problem:** Represents issues or challenges a system aims to resolve.
  - **Experiment:** Involves investigations on explorations or experimental designs of specific scenarios.
- **Research Study:** Represents structured investigations or studies conducted to gain insights or validate theories.
- **Human:** Includes entities such as :Student, :Stakeholder, :Teacher and others as shown in figure 6d, these classes represent the different roles humans take while interacting with the system, from general user groups, such as SystemUser, to specific domain user groups, such as Legal and Healthcare, or humans involved or interested in the domain such as :Researcher, :AIPractitioner, :DomainExpert, etc. The ontology also models human-to-human relationships, which further elucidates how individuals—though not direct users of the system—can still be significantly affected by it. Examples include patients whose data and outcomes are mediated by medical practitioners, or suspects whose legal processing is mediated by legal practitioners.
- **Input and Output:** Represents the information flow of multimodal datatypes.
- **Discipline and Domain:** These are made to be equivalent classes in the ontology. :Discipline represent study majors, while :Domain also includes other professional settings.

The objective of the integrated hybrid system is to develop an enhanced ontology structure that provides a more comprehensive domain-knowledge representation, for deeper semantic analysis.

2.2 Instance Creation

For the process of instance creation, specific terminologies are retrieved from the research papers under the specified ontological classes both manually and in hybrid versions. In the manual setups, the terms are identified based on the conceptualization available in the paper's context. In the hybrid methodologies, large language models, specifically GPT-04-Turbo known for its categorization abilities [3], are used to retrieve terms from a targeted research paper, through named entity recognition techniques, such that the terms retrieved relate to the existing classes in the ontology design. Further, the instances are verified again to avoid repetition of the same terminology from the manual extraction.

The Figure 6a explains the distribution of individuals declared under different sets of classes and subclasses. Among others, the system and capability classes have the majority of individuals under them.

The total number of individuals per concept is illustrated in the Figure 5. It is evident that :System represents approximately 50%, followed by the :Scenario and Tasks classes.

2.3 Linking

DBpedia offers a widely adopted, multilingual, and semantically rich ontology of real-world concepts. Linking to DBpedia allows us to place our domain-specific classes in context with the broader Linked Open Data cloud, enabling enhanced reasoning, query federation, and interoperability across applications.

The goal of linking our ontology to DBpedia is to enrich our classes with global semantics and make them reusable and integrable in online knowledge infrastructure. As Parundekar et al. (2010) explain, ontology alignment allows one source to be interpreted in terms of another, facilitating concept-level interoperability across heterogeneous systems [16].

Linking to external ontologies involves risks such as semantic mismatch or overcommitting to equivalence. To address these:

- We used `owl:equivalentClass` for clearly aligned concepts (e.g., `University`, `ResearchStudy`, `System`, `Subject`).
- We used `rdfs:subClassOf` for context-specific or narrower roles (e.g., `Judge`, `DefenceAttorney`).
- Through selective equivalence and subclassing, we balance semantic alignment with structural control.

HI Ontology–DBpedia Class Integration Mapping The following are the proposed mappings between the classes in our ontology and corresponding concepts in the DBpedia ontology:

- `Case` →`dbo:Case`
  Mapped using `owl:equivalentClass`. Our ontology modeled `Case` as a subclass of `LegalCase`, which conflicts with DBpedia's structure, where `LegalCase` is a subclass of `Case`. The class hierarchy in our ontology must be matched to DBPedia first,to ensure semantic and structural equivalence, and accordingly safe linking.

  ```
  :Case owl:equivalentClass dbo:Case .
  ```

- `Judge` →`dbo:Judge`
  Integrated using `rdfs:subClassOf`. While DBpedia models `dbo:Judge` primarily as an occupation under `dbo:Person`, our ontology treats `Judge` as a functional role within legal scenarios. Using subclassing allows us to preserve this procedural nuance without semantic conflict.

  ```
  :Judge rdfs:subClassOf dbo:Judge .
  ```

- `DefenceAttorney` →`dbo:Lawyer`
  Mapped via `rdfs:subClassOf`. Similarly,`DefenceAttorney` represents a specialized legal role, which is logically a subset of the broader `dbo:Lawyer` category in DBpedia.

  ```
  :DefenceAttorney rdfs:subClassOf dbo:Lawyer .
  ```

- `Subject` →`dbo:AcademicSubject`
  Linked using `owl:equivalentClass`.

  ```
  :Subject owl:equivalentClass dbo:AcademicSubject .
  ```

- `System` →`dbo:Software`
  Mapped through `owl:equivalentClass`.our `Subject` conforms with DBpedia's definition, representing academic subjects such as physics and computer science.

  ```
  :System owl:equivalentClass dbo:Software .
  ```

DBPedia has a total of 263,464 instances, for all equivalent classes to our ontology. The below Figure 4 shows the instance distribution by class.

3 Insights

3.1 Querying and Visualizing with SPARQL

Count of classes and their instances Figure 5 shows all defined classes and the frequency of individuals that belong to each, showing the collective occurrence of classes in the data.

```
SELECT ?class (COUNT(?instance) AS ?count)
WHERE {
  ?instance a ?subclass .
  ?subclass rdfs:subClassOf* ?class .
}
GROUP BY ?class
ORDER BY DESC(?count)
```

We further extract the core concepts in the ontology, and look at their instance distribution.

Count of instances of (classes + their subclasses) Figure 6 shows the design of core concepts in the ontology: Systems (including their capabilities, characteristics, tasks and scenarios), and humans.

```
SELECT ?class (COUNT(?instance) AS ?count)
WHERE {
  ?instance a ?subclass .
  ?subclass rdfs:subClassOf* ?class .
}
GROUP BY ?class
ORDER BY DESC(?count)
```

Extract all Triples Figure 7 shows the frequency of relationships (properties) appearing between instances in the ontology triples.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>

SELECT ?subject ?predicate ?object
WHERE {
  ?subject a ?class1 .
  ?object a ?class2 .
  ?subject ?predicate ?object .

  FILTER (
    !STRSTARTS(STR(?subject), STR(rdf:)) &&
    !STRSTARTS(STR(?subject), STR(rdfs:)) &&
    !STRSTARTS(STR(?subject), STR(owl:)) &&
    !STRSTARTS(STR(?object), STR(rdf:)) &&
    !STRSTARTS(STR(?object), STR(rdfs:)) &&
    !STRSTARTS(STR(?object), STR(owl:)) &&
    !STRSTARTS(STR(?predicate), STR(rdf:)) &&
    !STRSTARTS(STR(?predicate), STR(rdfs:)) &&
    !STRSTARTS(STR(?predicate), STR(owl:))
  )
}
```

3.2 Link Prediction and Knowledge Graph Completion

To assess the potential for enriching the *hi_ontology* knowledge graph, we conducted a link prediction task using the PyKEEN framework. The knowledge graph comprised 398 triples and 292 unique entities. The data was split into

85% training and 15% testing sets. We trained a ComplEx embedding model for 200 epochs using pairwise hinge loss, LP regularization, and negative sampling.

Model performance was evaluated using standard ranking metrics. The ComplEx model achieved a Mean Reciprocal Rank (MRR) of 0.05, a Hits@10 of 10%, and a Hits@3 of 0%, indicating low to moderate predictability. The low MRR and Hits@3 can be attributed to the sparsity and structural complexity of the ontology, which included 123 distinct relations.

To verify the model's potential for graph completion, we performed batch scoring on unseen triples. For every candidate (h, r, t) not already present in the knowledge graph, the model assigned a plausibility score. The top predicted triples are shown in Table 1 and Figure 8

Table 1: Top predicted triples generated by the ComplEx model.

Head	**Relation**	**Tail**
AccountabilityLossConsequence	ns1:involves	AccountabilityLossConsequence
AssessmentAndEvaluationApplication	ns1:usesModel	AssessmentAndEvaluationApplication
AssessmentAndEvaluationApplication	ns2:inScenario	AssessmentAndEvaluationApplication
ComplexityScientist	ns2:inScenario	ComplexityScientist
AlibiStatement	ns1:assessesCapability	AI_usability

The raw plausibility scores ranged from 129 to 171, with higher scores indicating a greater likelihood of the triple being semantically valid. These scores, while not probabilistic, provide a useful ranking of candidate links. These results illustrate the model's capacity to surface structurally plausible but unasserted relationships, which can assist domain experts in ontology refinement and extension. Future work could integrate domain-specific constraints for better outcomes.

4 Research Proposal

4.1 Problem Definition

Manual ontology extraction is often labor-intensive, inconsistent, and not scalable for large-scale academic corpora. On the other hand, fully automated ontology extraction using large language models (LLMs) can lack semantic precision and domain adaptation, and often produces structurally inconsistent ontologies. Our research addresses the challenge of developing an optimal, hybrid extraction pipeline that combines the nuanced reasoning capabilities of humans with the speed and scale of machine intelligence. The core problem is to identify how tasks should be divided and aligned between human annotators and LLMs to maximize semantic quality, structural consistency, and scalability of ontology generation from scholarly papers.

4.2 Motivation

With the advent of academic research in different disciplines, knowledge structuring becomes necessary. The development of a hybrid model bridges the gap between labor-intensive manual extraction along with domain-specific inconsistencies among Large Language Models. This can be executed with the assignment of domain-specific basic structuring to humans and the task of extraction and system integration suggested by the Large Language Models.

5 Related Work

Recent advances in ontology extraction and knowledge graph (KG) construction have explored both symbolic and neural approaches, often with varying degrees of human involvement. Borisova [6] presents a rule-based pipeline using the GATE framework and an OWL-Lite ontology, leveraging linguistic preprocessing (lemmatization and POS tagging) to map Bulgarian language tokens to formal concepts. While this method achieved high accuracy (287 out of 290 terms matched with lemmatization), it was limited by language dependency and lacked human-in-the-loop feedback. This highlights a common issue in rule-based systems: high precision at the cost of flexibility and generalizability.

Sanagavarapu et al. [19] take a different direction with OntoEnricher, a deep learning pipeline using LSTM models to enrich existing ontologies. They extracted over 97,000 RDF triples from DBpedia, which were manually filtered by experts to train the system. While the model performed well in structured domains like food (high F1-scores), it struggled with more complex domains such as cybersecurity. This approach underscores the scalability of neural models but also illustrates the limitations of fully automated pipelines in handling semantic ambiguity and domain complexity without dynamic human oversight.

Abolhasani and Pan [1] propose a middle ground with OntoKGen, an iterative Chain-of-Thought (CoT) based pipeline that employs Large Language Models (LLMs) for KG generation. The system is guided by users through an interactive interface, allowing for domain-specific customization and adaptive learning. OntoKGen exemplifies how LLM-driven automation can be enhanced with structured human interaction, enabling both semantic control and automation — a principle central to our own hybrid ontology extraction approach.

In the engineering domain, the Tunnel2024 study [25] introduces a hybrid NLP-based method tailored to rock tunnel support design. Their method integrates rule-based extraction (from technical standards) with statistical methods (applied to scientific papers), resulting in a domain-specific KG that captures both classification and non-classification relationships. Although no quantitative evaluation is reported, the method successfully extracted 947 entities and 14 relationship types, demonstrating the feasibility of hybrid NLP-KG pipelines in specialized contexts.

Lastly, Javid and Shah [11] propose a symbolic-neural hybrid model designed for scalable knowledge map construction across domains such as healthcare and finance. Their architecture combines symbolic logic with machine learning to improve transparency, accuracy, and real-time updating. While this system excels in integrating reasoning and learning, it faces challenges in aligning symbolic rules with neural predictions and in minimizing bias from training data.

Several tools and frameworks have recently explored hybrid or generative approaches to ontology enrichment and knowledge graph construction. *RDF-Hunter* [2] is a hybrid query engine that enhances SPARQL query completeness by combining automated RDF querying with crowd-sourced data completion tasks. It uses a quality model to assess data completeness and uncertainty, then dynamically offloads incomplete triple patterns to the crowd, yielding strong F-measure scores (0.84–0.96) across five domains. However, the system depends heavily on human input and struggles with knowledge areas where crowd familiarity is low.

In parallel, industry-facing tools offer practical solutions to ontology structuring and enrichment. **AutoRDF2GML** [18] converts RDF data into formats suitable for graph machine learning, but its performance is limited when RDF input is poorly structured. **ChatGPT and generative AI models** [13] are being applied to generate taxonomies and relationships, but often result in inconsistent hierarchies and missing reciprocal relations, necessitating manual refinement. Similarly, **Generative AI for Ontology Development** [21] suggests concepts and links based on domain-specific corpora, but struggles with nuanced domain understanding and still requires expert supervision for semantic quality control.

These tools reflect a common trade-off in current ontology development pipelines: automated systems excel in scale and speed but often fall short in precision and contextual depth. Our proposed solution builds on the insights from these works. We present, in this paper, a hybrid workflow that uses LLMs to automate candidate extraction while preserving human reasoning in curation, validation, and semantic structuring. This approach aims to resolve some of the limitations presented in previous systems, by explicitly balancing automation with human interpretability, particularly in multi-domain corpora grounded in explainability, ethics, and human-machine collaboration.

5.1 Research Hypothesis and Research Questions

The research investigates the optimization of human and artificial intelligence collaboration to capture semantic nuance and structural consistency in the construction of ontologies. To direct this inquiry, the following sub-research questions are proposed:

- **RQ1.** *How much does the hybrid extraction pipeline enhance semantic accuracy and minimize inconsistencies in comparison to entirely automated methods with the help of confidence scores and human-guided pre-processing?*
- **RQ2.** *What is the effect of hybrid extraction on the scalability of ontology generation for large research corpora, particularly concerning time efficiency, resource utilization, and manual effort as dataset size increases?*

To effectively examine the following inquiries, the hypothesis below has been formulated.

- **Hypothesis:** *A hybrid ontology extraction pipeline that systematically allocates tasks between human annotators and Large Language Models (LLMs) enhances the semantic precision, structural coherence, and scalability of ontologies derived from literature, in comparison to approaches that are entirely manual or fully automated.*

The hypothesis explores the advantages of combining human input with advanced computational models in the ontology extraction process, focusing on the potential improvements offered by a hybrid approach in enhancing the quality of ontologies derived from literature.

5.2 Proposed Approach and Contribution

The system flow architecture, as illustrated in Figure 9, consists of six primary components that streamline ontology development. The initial data sources include both research papers and an existing system ontology that we aim to expand. A subset of the target research papers were selected for ontology conceptualization through manual knowledge engineering processes. The remaining research papers undergo preliminary data preprocessing to convert them into a suitable format. Given that LLMs excel in understanding terms in their textual form without extensive preprocessing that might alter the semantics, the textual data is directly fed into the model for subsequent natural language processing tasks.

The Named Entity Recognition (NER) module identifies key entities and instances from the extracted text of the research papers, relying on the classes defined in the manually extracted ontology. For each identified term, a confidence score is generated, prompting the LLM to critically assess the outcomes. The instances retrieved are then subjected to a process of elimination to filter out redundant terms with the existing ontology, thereby minimizing potential discrepancies in later stages.

In the semantic relationship extraction module, the extracted text and the current set of individuals are transformed into word embeddings using FastText [14]. The similarity between words is evaluated using cosine similarity metrics to establish semantic relationships between entity pairs.

Terms that demonstrate a high similarity score are considered for relationship definition, with assistance from the LLM. The model first assesses the feasibility of utilizing existing relationships between the terms before opting for new ones. This approach reduces conflicts arising from multiple similar relationships, thereby enhancing the final ontological development.

The relationships identified between existing and new individuals, as well as those among the new individuals themselves, form the basis of the final relationships. These relationships, along with the confirmed individuals, are integrated into the existing ontology.

5.3 Experimental Evaluation

For each NLP task, both the automated and hybrid versions are evaluated:

NER - Named Entity Recognition: The fully automated model was used In the task of NER, to extract terms from the research paper directly without any structured information about the final ontology. The model identified terms based on the existing classes in the initial ontology where there were no specific instructions about how to extract the terms. We observed model hallucinations, that resulted in duplicates, contradictions, and out of scope relationships - see Figure 10a. The final outcome was an inconsistent and non representative ontology. In the hybrid model, specific instructions were provided based on the expected ontology framework. Here, duplicates were further reduced with preprocessing measures. Additionally, prompting the model included explicit confidence score intervals shown in Figure 10c. This serves for more concrete model evaluation, and reduces risks of overfitting and underfitting.

Semantic Relationship Extraction: The fully automated model was designed to establish direct relationships between newly retrieved terms and existing properties. However, inconsistencies arose as the model attempted to form connections between retrieved terms (representing new terms) and existing classes instead of established instances, as mentioned in Figure 10b. This posed a complexity for consistency evaluation for the constructed ontology. In contrast, the hybrid model resolves this challenge through employing cosine similarity metrics to assess the similarity scores of the involved terms. Separate data files were created to outline relationships between pairs of new terms, new terms and existing ones that were manually, as shown in Figure 10d and Figure 10e. To prevent redundancy in relationships, which was observed in the fully automated process, we factored an initial step for the model to evaluate similarity scores between terms generated by the model and existing property terms and only assert new connections.

6 Conclusion

The notion of hybrid intelligence—combining human expertise with artificial intelligence—can be meaningfully extended to the domain of ontology design and development. In this context, Large Language Models (LLMs) can be repurposed as ontology engineering assistants. They have the potential to reduce manual overhead, accelerate modeling tasks, and improve efficiency in generating ontological structures.

Despite their remarkable fluency and ability to simulate reasoning patterns, current LLMs are fundamentally designed for next-word prediction. This often leads to limitations in their ability to capture semantic abstractions or produce ontologies with deep conceptual coherence [15]. As such, LLMs are not a replacement for ontology engineers but can serve as powerful co-pilots when guided appropriately.

Repurposing LLMs for ontology engineering involves a structured, semi-automated process, typically comprising the following steps:

1. Initial ontology scoping and high-level design by a human expert.
2. Prompt-driven generation of classes, properties, constraints, and restrictions.
3. Iterative refinement and validation, ideally supported by reasoning tools.
4. Final evaluation and review, including human-in-the-loop assessment and quality control.

Looking forward, future advancements should focus on enhancing prompting strategies to produce more semantically consistent and less redundant output. Moreover, LLM integration with ontology engineering environments—such as plugins for tools like Protégé—could enable more seamless, interactive workflows. The goal is to foster a relationship in which LLMs handle repetitive or structural tasks, while human experts focus on conceptual accuracy, reusability, and domain alignment.

References

[1] Mohammad Sadeq Abolhasani and Rong Pan. "OntoKGen: Leveraging LLMs for Automated Ontology Extraction and Knowledge Graph Generation". In: *arXiv preprint arXiv:2412.00608* (2024). URL: `https://arxiv.org/abs/2412.00608`.

[2] Maribel Acosta et al. "RDF-Hunter: Automatically Crowdsourcing the Execution of Queries Against RDF Data Sets". In: *Semantic Web. Latest Advances and New Domains - ESWC 2015 Satellite Events*. Springer, 2015, pp. 448–453. URL: `https://arxiv.org/abs/1503.02911`.

[3] Clarissa Arlinghaus et al. *From GPT to Open-Source LLMs: Advancing LAIC for More Efficient Qualitative Analysis*. Mar. 2025. DOI: `10.13140/RG.2.2.15049.25442`.

[4] Julian Bauer and Frank van Harmelen. "Ethical Preferences in the Digital World: The EXOSOUL Questionnaire". In: *Journal of AI Ethics* (2022).

[5] Eva Blomqvist and et al. "Open, Multiple, Adjunct. Decision Support at the Time of Relational AI". In: *Semantic Web Journal* (2022).

[6] Nadezhda Borisova. "An Approach for Ontology-Based Information Extraction". In: *International Conference on Computational Linguistics and Intellectual Technologies (Dialogue)*. 2018, pp. 77–85.

[7] Adolfo Guzmán-Arenas and Jorge I Toledo. "Automatic Building of an Ontology from a Corpus of Text Documents Using Data Mining Tools". In: *Computación y Sistemas* 18.3 (2014), pp. 485–499.

[8] Frank van Harmelen and Annette ten Teije. "Knowledge Graphs in Support of Human-Machine Intelligence". In: *Lecture Notes in Computer Science (LNCS)*. Vol. 13469. Springer, 2022, pp. 1–12. DOI: `10.3233/SSW220012`.

[9] Dirk Helbing. "The Games We Play: How Critical Complexity Improves Machine Learning". In: *AI Society* (2023).

[10] Marcello Ienca and Effy Vayena. "Legal-Ethical Challenges and Technological Solutions to Health Data Consent in the EU". In: *Health Data Science* (2021).

[11] Eman Javid and Wasif Shah. "Hybrid AI Models for Large-Scale Information Extraction and Knowledge Map Construction". In: *ResearchGate Preprint* (2025). Available online: `https://www.researchgate.net/publication/388526633`.

[12] Aditya Khosla and et al. "Egocentric Hierarchical Visual Semantics". In: *CVPR* (2023).

[13] Enterprise Knowledge. *Generative AI for Taxonomy Creation: Opportunities and Risks*. `https://enterprise-knowledge.com/generative-ai-for-taxonomy-creation/`. Accessed April 2025. 2024.

[14] Shilpi Kulshretha and Lokesh Lodha. "Performance Evaluation of Word Embedding Algorithms". In: *International Journal of Innovative Science and Research Technology* 8 (Dec. 2023), pp. 1555–1561. DOI: `10.5281/zenodo.10443962`.

[15] Melanie Mitchell. *Can Large Language Models Reason?* `https://melaniemitchell.me/2023/09/14/can-large-language-models-reason`. Accessed: 2025-04-07. 2023.

[16] Rahul Parundekar, Craig A Knoblock, and José Luis Ambite. "Linking and building ontologies of linked data". In: *The Semantic Web–ISWC 2010: 9th International Semantic Web Conference, ISWC 2010, Shanghai, China, November 7-11, 2010, Revised Selected Papers, Part I 9*. Springer. 2010, pp. 598–614.

[17] Gunes Plumb, Adler Perotte, et al. "Exemplars and Counterexemplars Explanations for Skin Lesion Classifiers". In: *Proceedings of ICML 2021*. 2021.

[18] Restack.io Editorial Team. *Best Practices for RDF and AI Interoperability*. `https://www.restack.io/p/rdf-best-practices-answer-ai-data`. Accessed April 2025. 2024.

[19] Lalit Mohan Sanagavarapu, Vivek Iyer, and Raghu Reddy. "A Deep Learning Approach for Ontology Enrichment from Unstructured Text". In: *arXiv preprint arXiv:2112.08554* (2021). URL: `https://arxiv.org/abs/2112.08554`.

[20] Luisa Schulz and et al. "Unreflected Acceptance – Investigating the Negative Consequences of ChatGPT-Assisted Problem Solving in Physics Education". In: *Education and Information Technologies* (2023).

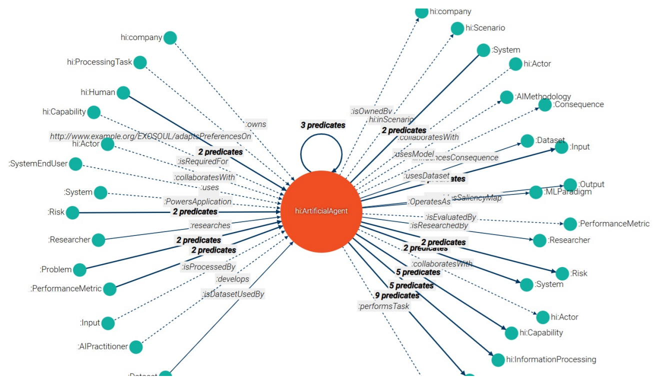
[21] Progress Software. *How Can Generative AI Create, Enrich and Manage Semantic Models*. `https://www.progress.com/blogs/how-can-generative-ai-create-enrich-and-manage-semantic-models`. Accessed April 2025. 2024.

[22] Ilaria Tiddi et al. "Knowledge engineering for hybrid intelligence". In: *Proceedings of the 12th Knowledge Capture Conference 2023*. 2023, pp. 75–82.

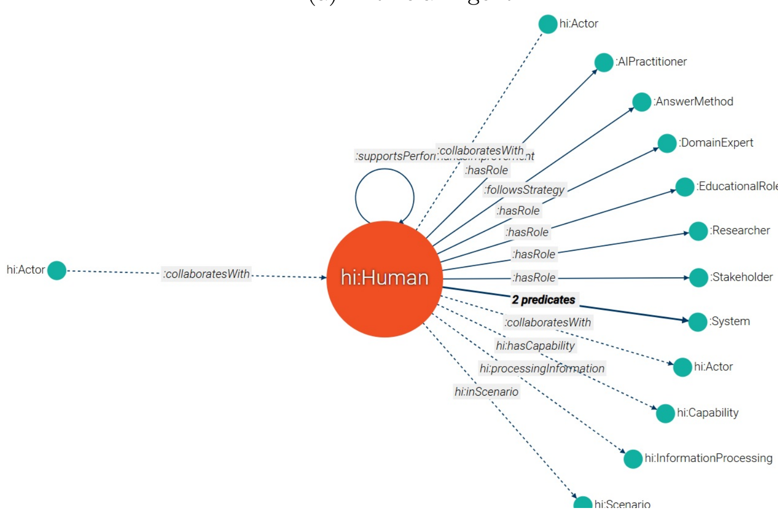
[23] Hans Timmer and Henry Prakken. "Building a Stronger Case: Combining Evidence and Law in Scenario-Based Bayesian Networks". In: *Proceedings of ICAIL*. 2020.

[24] Dan Wang and et al. "A Conceptual Model for Implementing Explainable AI By Design". In: *IEEE Access* (2020).

[25] Jie Wu et al. "A Hybrid NLP-based Extraction Method to Develop a Knowledge Graph for Rock Tunnel Support Design". In: *Automation in Construction* (2024). Available online: `https://www.sciencedirect.com/science/article/abs/pii/S1474034624003732`.

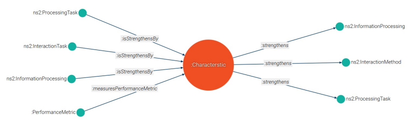


(a) Artificial Agent

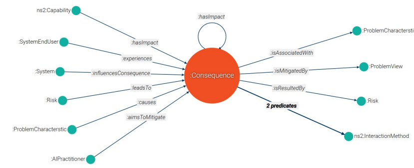


(b) Human

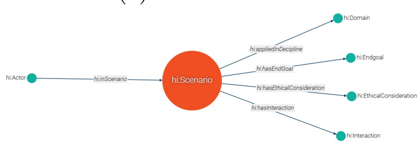
Fig. 2: **Ontology Classes Visualization:** Expanded ArtificialAgent and Human Classes.



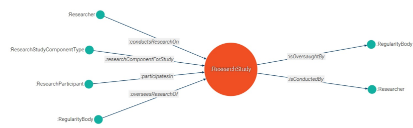
(a) Characteristic



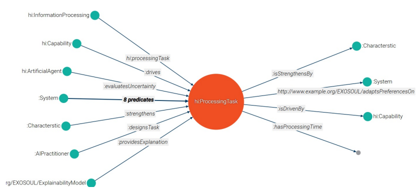
(b) Consequence



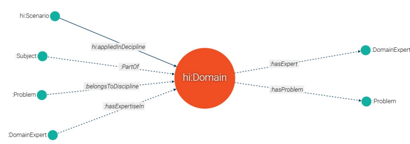
(c) Scenario



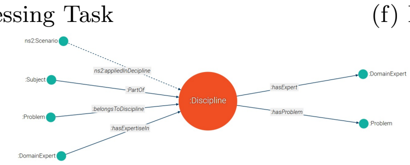
(d) Research Study



(e) Processing Task



(f) Domain



(g) Discipline

Fig. 3: **Ontology Classes Visualization:** New ontology classes such as Characteristic, Consequence, Discipline, Scenario, and Research Study, are represented as a separate sub-figures.

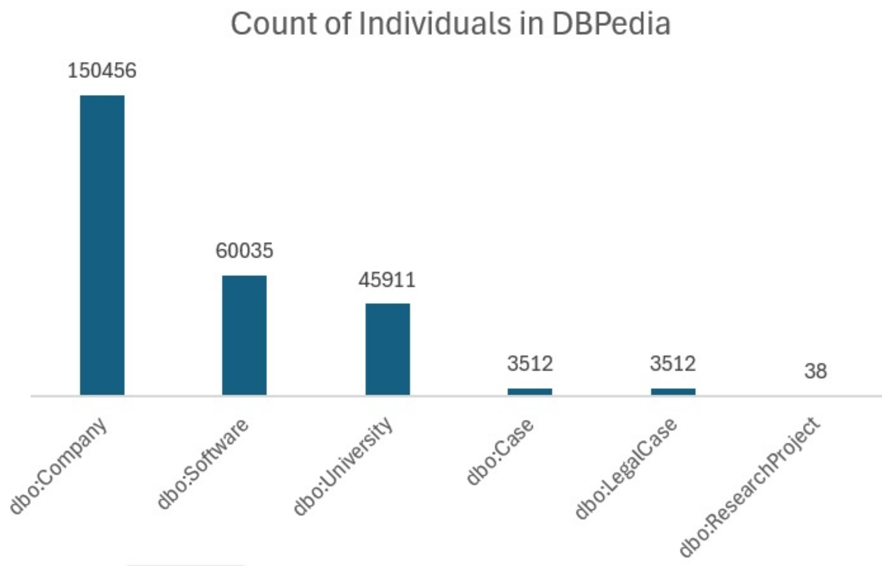


Fig. 4: DBPedia Instances for classes: company, software, university, Case, legal case

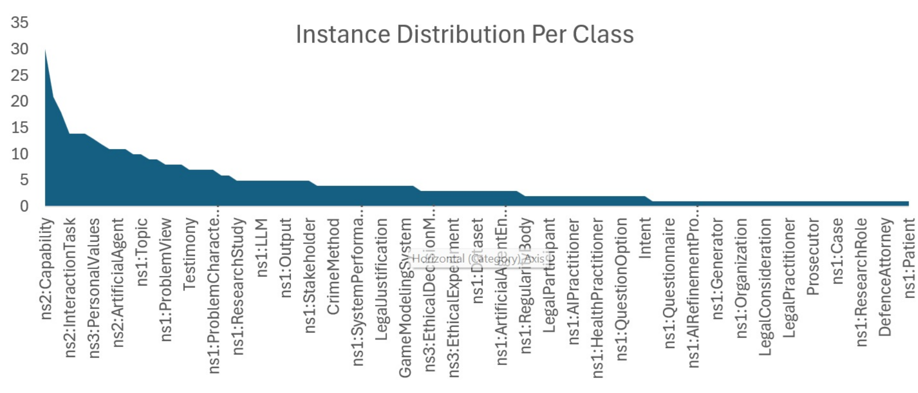
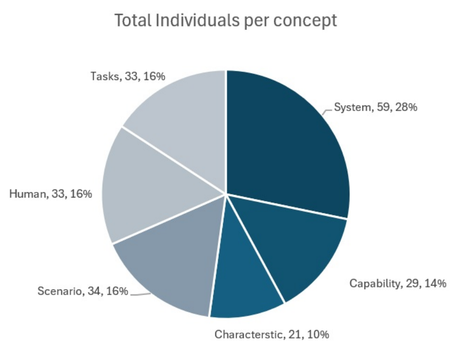
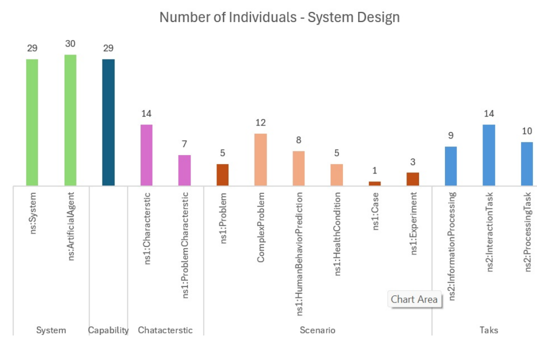


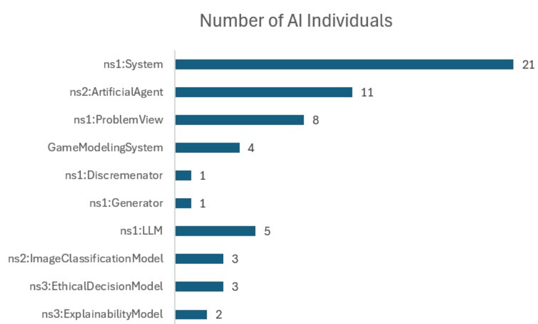
Fig. 5: Frequency of Instances per Class



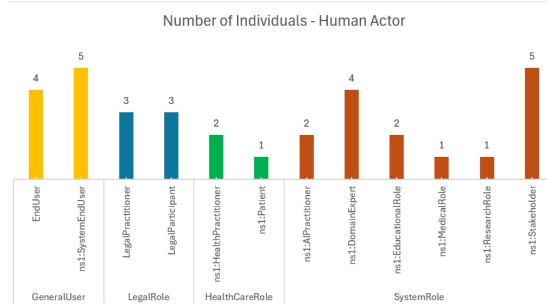
(a) Frequency of Instances per concept



(b) Distribution of instances for system design



(c) distribution of instances for system



(d) Distribution of instances for human actor

Fig. 6: **Instance Visualization:** Description on the instances under specific classes.

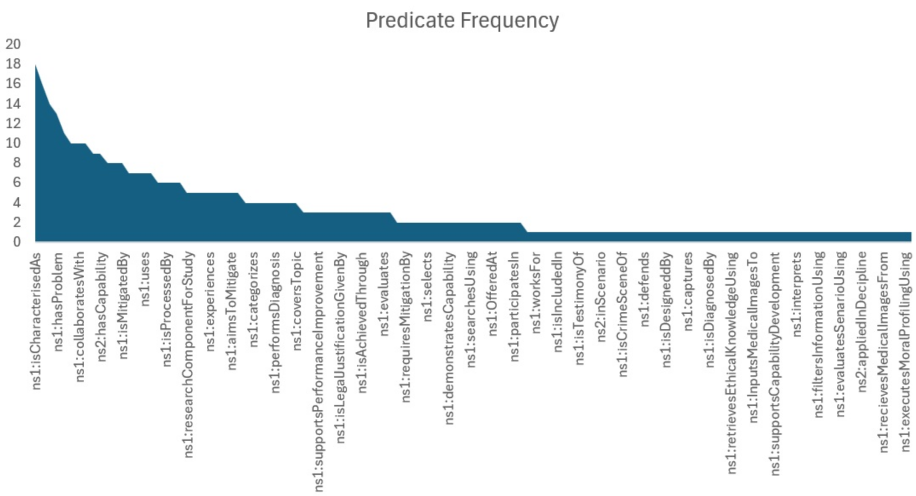
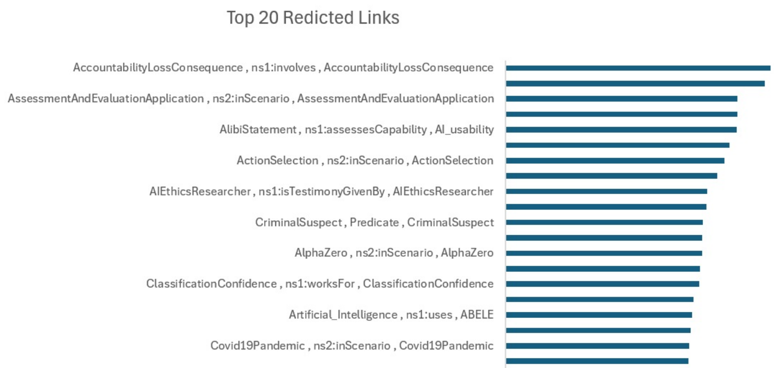
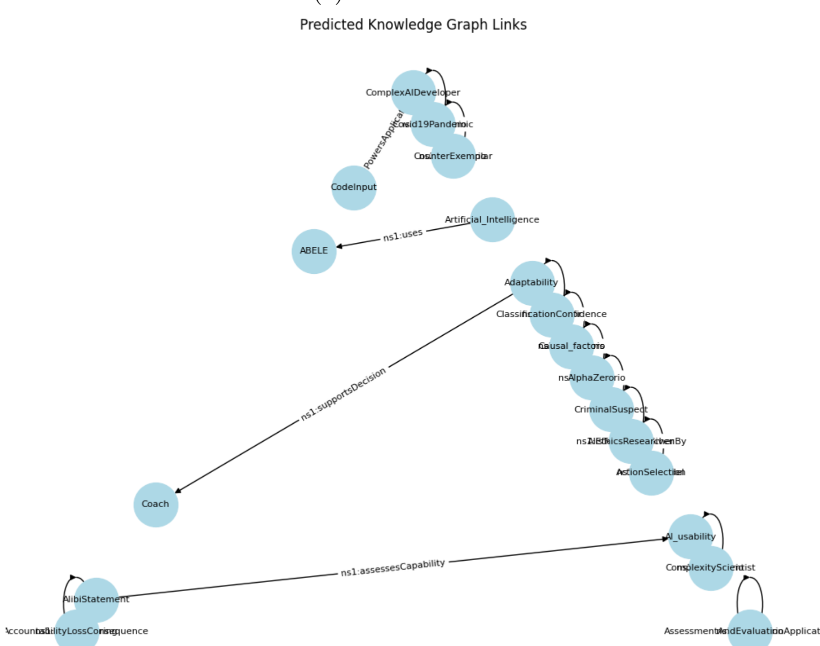


Fig. 7: Frequency of relationships (properties)



(a) Predicted links



(b) Missing Links Graph

Fig. 8: **Link Prediction:** Top 20 predicted links and their plausibility Scores for the Predicted Graph

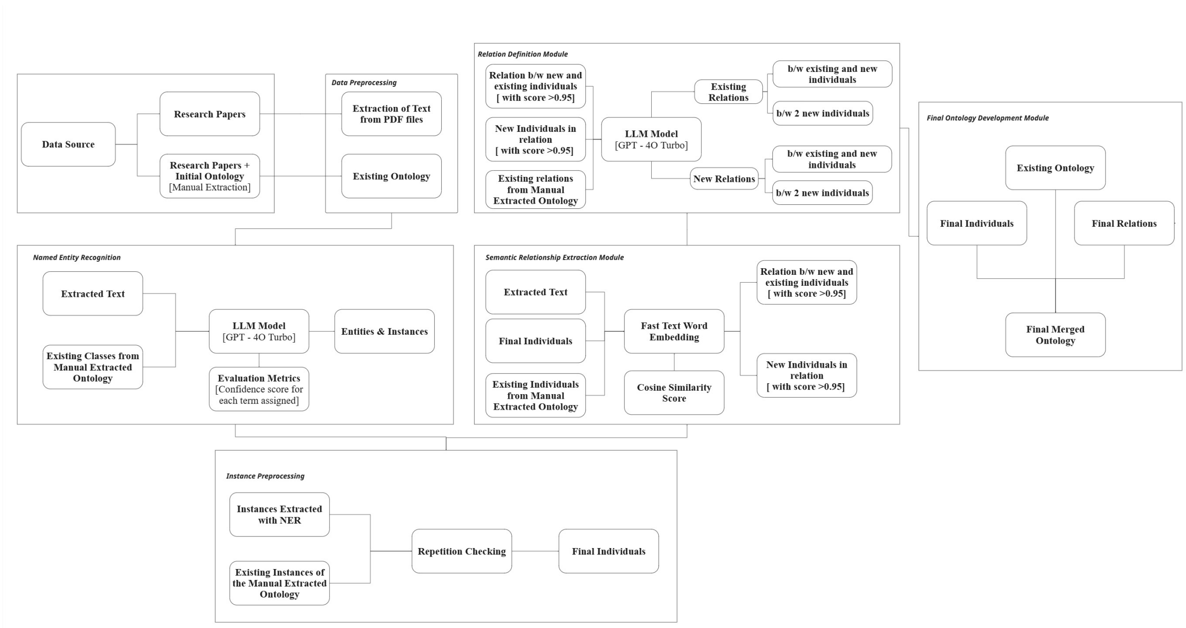
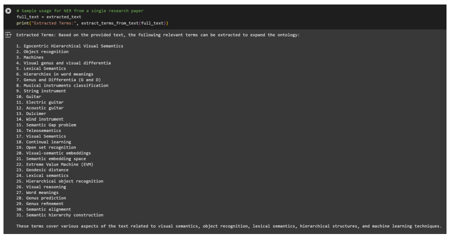
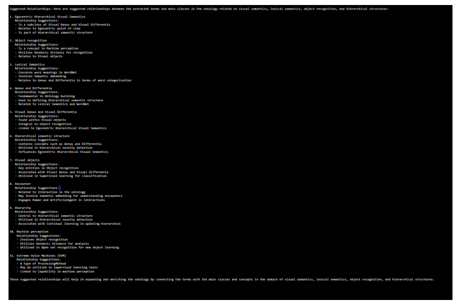


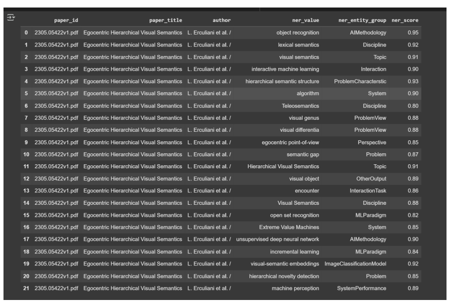
Fig. 9: Hybrid System Flow



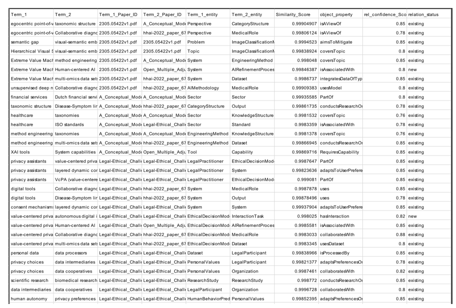
(a) Named Entity Recognition (Automated)



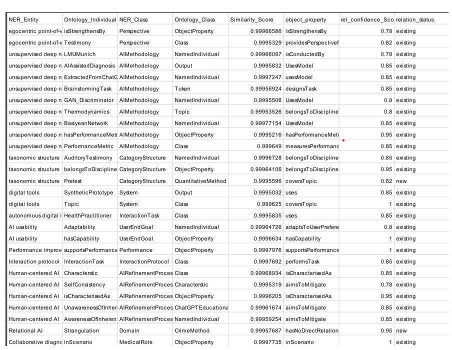
(b) Relation Prediction (Automated)



(c) Named Entity Recognition (Hybrid)



(d) Relation Prediction New-New terms (Hybrid)



(e) Relation Prediction New-Existing terms (Hybrid)

Fig. 10: Representations of extracted ontology relationships, NER outputs, and relation modeling from research paper.