SynSem-Align (Demo): Ontology-Driven KG Extraction via Syntactic Candidate Mining and Paraphrase-Based Equivalence Filtering

Rikuto Sasaki[1,†], Masahito Yasui[1,†] and Kazuhiro Takeuchi[1,*]

[1]Osaka Electro-Communication University

Abstract

While Large Language Models (LLMs) are powerful for information extraction, the reliability of their output remains a challenge, making human supervision essential. We introduce SynSem-Align, a support tool where LLMs and humans collaborate on knowledge extraction. Our approach integrates three core components: (1) Ontology-Driven filtering to suggest relevant extraction patterns, (2) Syntactic Candidate Mining to precisely identify knowledge candidates using a CKY-based approach over dependency structures, and (3) Paraphrase-Based Equivalence Filtering using an LLM for semantic validation. This integrated workflow enables users to transparently and reliably construct knowledge graphs, demonstrating a practical path towards verifiable knowledge extraction that balances automation with human oversight.

Keywords

Knowledge Graph, Ontology, Natural language processing, Large Language Model, Semantic analysis, Information extraction,

1. Introduction

While Large Language Models (LLMs) have significantly advanced automated knowledge extraction, ensuring the reliability and verifiability of their outputs remains a key challenge [1]. End-to-end extraction methods that rely solely on LLMs often lack transparency, making it difficult for human experts to verify results or correct errors. Because the internal reasoning of these models is not directly observable, guaranteeing its faithfulness remains challenging [2].

In this paper, we present SynSem-Align, a tool for verifiable, human-supervised knowledge extraction. SynSem-Align externalizes the extraction logic as explicit, selectable patterns, rather than relying on implicit, model-internal processing. In contrast to black-box approaches with LLMs that attempt to generate triples directly from prompts embedding both the source text and ontology constraints, our tool first enumerates multiple candidate triples and then refines them using ontology-based filtering and paraphrase-based equivalence checks. This combination ensures that extraction is not only guided by linguistic constraints but also accompanied by reasoning steps that can be inspected and corrected. A distinctive aspect of SynSem-Align is that it explicitly acknowledges inherent ambiguities: even after refinement, alternative knowledge representations may remain. Instead of suppressing this diversity, SynSem-Align provides mechanisms to expose it to human judgment. Through an interactive interface for visual verification, users can examine how textual spans map onto candidate triples and adjudicate among competing interpretations. This design allows knowledge graphs to be constructed in a manner that is transparent, auditable, and domain-adaptable.

To complement these methodological contributions, we provide a demonstration that walks through SynSem-Align's workflow step by step. To ensure reproducibility and foster further exploration, the source code is publicly available on GitHub[1].

ISWC 2025 Companion Volume, November 2–6, 2025, Nara, Japan

*Corresponding author.

†These authors contributed equally.

mi25a002@oecu.jp (R. Sasaki)

[1]https://github.com/TKLB-OECU/KG-Extraction-SynSem

2. One Text Span Admits Many Candidate Knowledge Graph Triples

A fundamental task in knowledge extraction is to make explicit, for a specific span of text, what knowledge can justifiably be extracted and why. In our view, the core difficulty is not a simple opposition between syntax and semantics but a residual gap between the results of syntactic and semantic analyses and the knowledge graph that we finally construct. Syntactic and semantic analyses constrain the space of possible meanings, yet the set of valid interpretations remains diverse, and that diversity is often domain dependent even for expressions that share the same surface structure.

Consider the metonymic phrase 'Land of Smiles'. The 'A of B' construction typically offers A and B as candidates for relations in the knowledge graph, for example attribution, origin, or possession; in this instance a natural interpretation is that the residents in A frequently smile (B). This illustrates that even text spans that share the same syntactic 'A of B' structure can yield diverse semantic relationships between their terms. Furthermore, even after applying both syntactic and semantic constraints, the mapping from the analyzed text to a knowledge graph may still be indeterminate. If we admit a latent variable X that does not appear in the span, the partial expression 'A of B' can license multiple candidate relations, for example $P(A, B)$, $A(X, B)$, $A(B, X)$, and others. This indicates that, despite the restrictions provided by syntax and semantics, a substantial set of candidate knowledge graphs remains. Since the appropriate choice can vary by domain or field, tools are needed that make the mapping from text to knowledge graph relations explicit and verifiable on the basis of consistent reasoning.

End to end LLMs may learn such correspondences implicitly, but their black box nature prevents us from verifying how a particular triple was licensed by a particular piece of text, and it prevents users from trusting or correcting the outcome. We therefore emphasize transparency and controllability. Users should see which part of the text supports which candidate relation, and what reasoning connects them. In practice, we operationalize this with paraphrase based validation anchored at the span: if a syntactic candidate can be rephrased into a natural and semantically equivalent sentence aligned with the same text span, it is retained; otherwise, it is discarded. This procedure enumerates the diversity of interpretations that remain after syntax and semantics have constrained the space, and it justifies each retained interpretation in a way that can be inspected, trusted, and corrected by humans.

3. Related Work

The dominant approach for knowledge extraction currently leverages the zero-shot or few-shot capabilities of Large Language Models (LLMs) [1]. While powerful, these end-to-end methods suffer from a lack of transparency: the reasoning process by which an LLM maps sentence structures to ontological relations is inaccessible to users. Several works have attempted to expose the model's intermediate reasoning, but the reliability of such explanations remains uncertain [2].

Our work takes a different stance by grounding the extraction process in a pre-constructed *pattern base* governed by humans. This pattern base is semi-automatically derived from text-to-graph datasets such as Text2KGBench [3], which provide systematically aligned text–triple pairs and thus a practical foundation for reusable extraction rules. In SynSem-Align, the LLM is not used for end-to-end triple generation. Its role is narrowly confined to auxiliary tasks such as paraphrase-based validation. As a result, the rationale for each extracted triple is not hidden within an opaque model but is anchored in explicit patterns that users can inspect and control. This ensures that the extraction process remains transparent and verifiable. This line of research extends our earlier work on domain-specific knowledge extraction [4] and its enhancement with generative models [5].

SynSem-Align enhances transparency and makes explicit the assumptions underlying LLM reasoning by explicitly controlling syntactic and semantic judgments, thereby supporting users in making final decisions regarding knowledge extraction. This stance stands in contrast to the Auto-KG Agent [6], which has been proposed as a multi tool framework where LLMs act as agents to invoke relation extraction systems such as REBEL [7] and KnowGL [8], integrate the extracted triples, and re rank them.

REBEL is a seq2seq model based on BART-large that linearizes entity mentions, types, and relations for end to end extraction, while KnowGL combines knowledge generation, fact ranking, and Wikidata linking through fine tuned language models. By orchestrating these components via an LLM agent, Auto-KG Agent pursues autonomy and aims to improve triple extraction accuracy, particularly for complex sentences or those involving negation, while minimizing direct human involvement. Whereas Auto-KG Agent seeks to reduce the human role in the extraction process, SynSem-Align establishes an alternative paradigm in which efficiency is carefully balanced with verifiability, ensuring that extracted knowledge remains interpretable and under explicit human control.

4. Transparent Human Supervised Workflow of SynSem-Align

SynSem-Align structures the extraction process into explicit, inspectable steps. The workflow systematically enumerates syntactic candidates, prunes implausible ones through ontology based and paraphrase based validation, and reserves the final decision for the human user. This staged design ensures that the diversity of possible interpretations is preserved, implausible ones are pruned, and the outcome remains transparent and auditable.

4.1. Enumerating a Broad Set of Syntactic Candidates

The process begins with dependency parsing, which identifies grammatical relations and decomposes the syntactic units(Figure 1). This analysis enables the system to process complex constructions such as coordination [9, 10], ensuring that conjunctive phrases are segmented into appropriate substructures.

The resulting structure feeds into a CKY-based charting algorithm [11], which combines sentence fragments and systematically matches them against the pre-constructed *pattern base* (Figure 2). The pattern base consists of reusable templates such as '[X1] is [Y1]' or '[X1] of [Y1],' which explicitly link surface syntax to potential semantic relations. Through this process, the system enumerates all triples that the surface syntax can license, anchored to explicit and reusable patterns rather than opaque model inferences.

The outcome is a broad but systematically organized set of syntactic candidates, which inherently includes both plausible and spurious interpretations. At this stage, the goal is exhaustive enumeration, leaving the task of discarding implausible candidates to subsequent refinement stages.

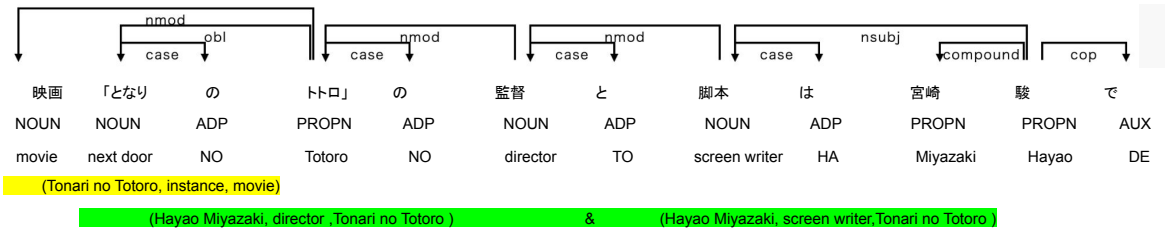

Figure 1: Dependency parsing provides the structural foundation for candidate generation.

4.2. Refining Candidates through Ontology and Paraphrase Validation

Because the syntactic candidate set is inherently diverse, it inevitably contains both plausible and spurious interpretations. This diversity is deliberately preserved, since subsequent refinement ensures that only semantically valid interpretations remain.

First, ontology based type and relation constraints rule out triples inconsistent with domain knowledge (Figure 3). For example, the system maps 'My Neighbor Totoro' to the concept `film` and 'director' to the relation `director`, discarding implausible triples such as 'film directed person'.

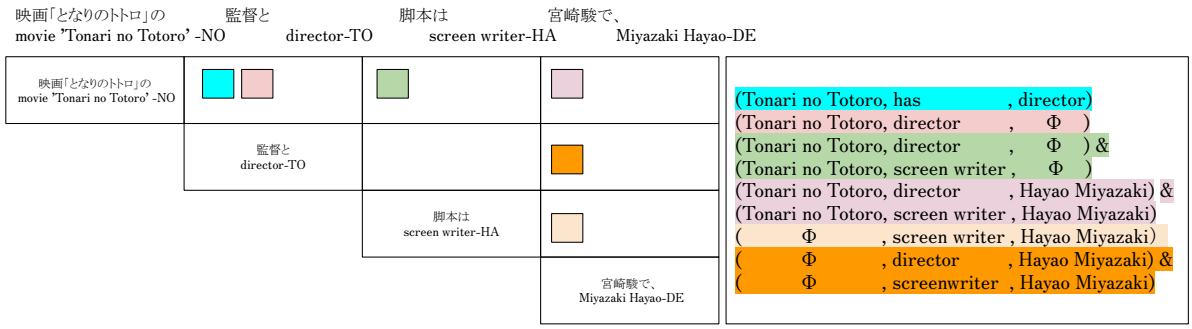

Figure 2: CKY-based matching with the pre-constructed pattern base enumerates multiple syntactic candidates.

Second, paraphrase based validation with an LLM evaluates the remaining candidates (Figure 4). Each candidate is rewritten into a normalized, simple declarative form and checked for equivalence with the original input.

This step does not generate new knowledge; instead, it excludes candidates that are structurally possible but semantically unnatural (e.g., 'a movie directed a person'). The LLM thus functions solely as an auxiliary consistency check positioned between syntactic generation and human selection.

Together, these mechanisms refine the syntactic candidate set into a semantically licensed subset, which is then presented for human inspection and final disambiguation.

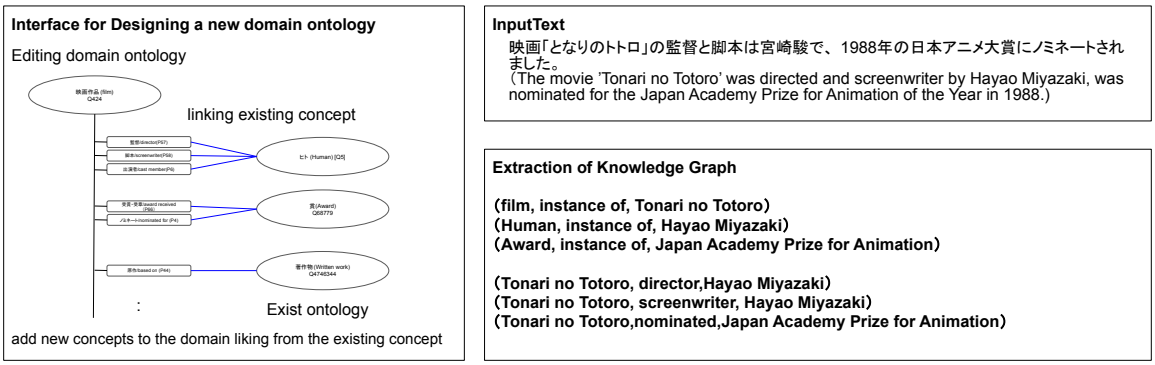

Figure 3: Workflow overview. Syntactic candidates are pruned by ontology constraints and paraphrase validation to yield semantic candidates.

4.3. Finalizing Knowledge through Human Guided Selection

Even after syntactic enumeration and refinement through ontology and paraphrase validation, multiple semantically valid candidates may remain. These reflect residual ambiguity that cannot be resolved automatically, and thus the decisive step is delegated to the human user.

The remaining candidates represented as abstract patterns such as '[X1] is [Y1]' or '[X1] of [Y1]' are presented in an explicit and inspectable form. The user visually inspects these alternatives and selects the pattern that best captures the intended meaning of the input sentence.

This explicit human decision directly determines the extracted knowledge and ensures that the reasoning process remains auditable: users can see which syntactic candidates were generated, which were excluded, and why the final pattern was chosen. The division of responsibilities is therefore clear: the system enumerates and refines candidates, the LLM provides only bounded auxiliary validation,

Purpose:
Rigorously verify the following two points about the parallel elements in the target sentence and output only “True” or “False” in JSON format.

Input:
- movie ’tonari no Totoro’ -NO director-TO screen writer-HA Miyazaki Hayao-DE
- Parallel elements: “director”, “screen writer” (e.g., "A", "B", "C")

Judgment Criteria:
1. Similarity Check:
- The head (central word) of each parallel element must be the same part of speech (e.g., all nouns or all verbs).
- Phrase structures should be aligned as much as possible (e.g., all “noun + particle”).

2. Commutability Check:
- Swapping the order of the parallel elements must still produce a natural Japanese sentence.
- Swapping only the parallel elements should not significantly change the overall meaning of the sentence (the main roles/semantic structure are preserved).

Decision Rule:
- Output “True” only if both criteria are satisfied, output “False” if either criterion is not met.
- The output must be **only** the following JSON format. Do **not** include any explanations or comments.

Figure 4: Paraphrase-based validation: a candidate is turned into a canonical sentence and checked for semantic equivalence to the input.

and the final disambiguation rests with the human expert. In this way, SynSem-Align guarantees that knowledge extraction remains transparent, controllable, and accountable, with interpretive authority explicitly assigned to the user.

5. Conclusion

We have presented SynSem-Align, a human supervised knowledge extraction tool. By effectively combining an ontology driven filter, a transparent syntactic matching engine based on dependency parsing and CKY, and LLM assisted semantic validation, our tool offers a transparent and verifiable workflow for supervised extraction.The explicit, pattern based logic, controlled by the user and verified through semantic checks, provides a robust framework for building reliable knowledge graphs, successfully balancing automation with essential human oversight.

Declaration on Generative AI

ChatGPT (OpenAI, 2025) was used only for English language checks (rephrasing, grammar, and style). All scientific content was created by the authors, who take full responsibility for the final manuscript.

Acknowledgment

This research was partially supported by JSPS KAKENHI Grant Number JP23K28152.

References

[1] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, in: Advances in Neural Information Processing Systems 33 (NeurIPS 2020), 2020.

[2] T. Korbak, M. Balesni, E. Barnes, Y. Bengio, J. Benton, J. Bloom, M. Chen, A. Cooney, A. Dafoe, A. Dragan, S. Emmons, O. Evans, D. Farhi, R. Greenblatt, D. Hendrycks, M. Hobbhahn, E. Hubinger, G. Irving, E. Jenner, D. Kokotajlo, V. Krakovna, S. Legg, D. Lindner, D. Luan, A. Mądry, J. Michael, N. Nanda, D. Orr, J. Pachocki, E. Perez, M. Phuong, F. Roger, J. Saxe, B. Shlegeris, M. Soto, E. Steinberger, J. Wang, W. Zaremba, B. Baker, R. Shah, V. Mikulik, Chain of thought monitorability: A new and fragile opportunity for ai safety, 2025. URL: https://arxiv.org/abs/2507.11473. arXiv:2507.11473.

[3] N. Mihindukulasooriya, S. Tiwari, C. F. Enguix, K. Lata, Text2kgbench: A benchmark for ontology-driven knowledge graph generation from text, in: Proceedings of the 22nd International Semantic Web Conference (ISWC 2023), 2023, pp. 247–265. doi:10.1007/978-3-031-47243-5_14.

[4] M. YASUI, T. KITAJIMA, H. TANIDA, K. MIYOSHI, S. OBA, K. TAKEUCHI, K. KOZAKI, Preprocessing for extracting knowledge from job and skill definition sentences with a simplified sentence pattern description, Proceedings of the Annual Conference of JSAI JSAI2025 (2025) 2Win524–2Win524. doi:10.11517/pjsai.JSAI2025.0_2Win524.

[5] M. YASUI, K. TAKEUCHI, Enhancing a tool for extracting knowledge graphs from text through the utilization of generative language models, JSAI Technical Report, Type 2 SIG 2024 (2024) 04. doi:10.11517/jsaisigtwo.2024.SWO-063_04.

[6] A. Ananya, S. Tiwari, N. Mihindukulasooriya, T. Soru, Z. Xu, D. Moussallem, Towards harnessing large language models as autonomous agents for semantic triple extraction from unstructured text, in: Extended Semantic Web Conference, 2024.

[7] P.-L. Huguet Cabot, R. Navigli, REBEL: Relation extraction by end-to-end language generation, in: M.-F. Moens, X. Huang, L. Specia, S. W.-t. Yih (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2021, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 2370–2381. URL: https://aclanthology.org/2021.findings-emnlp.204/. doi:10.18653/v1/2021.findings-emnlp.204.

[8] G. Rossiello, M. F. M. Chowdhury, N. Mihindukulasooriya, O. Cornec, A. M. Gliozzo, Knowgl: Knowledge generation and linking from text, in: AAAI, AAAI Press, 2023, pp. 16476–16478.

[9] Y. Sawada, T. Wada, T. Shibahara, H. Teranishi, S. Kondo, H. Shindo, T. Watanabe, Y. Matsumoto, Coordination boundary identification without labeled data for compound terms disambiguation, in: Proceedings of the 28th International Conference on Computational Linguistics (COLING 2020), International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 3043–3049. URL: https://aclanthology.org/2020.coling-main.271/. doi:10.18653/v1/2020.coling-main.271.

[10] H. Teranishi, H. Shindo, Y. Matsumoto, Decomposed local models for coordinate structure parsing, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for

Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 3394–3403. URL: https://aclanthology.org/N19-1343/. doi:10.18653/v1/N19-1343.

[11] T. KASAMI, An efficient recognition and syntax analysis algorithm for context-free languages, Science Report, Air Force Cambridge Research Laboratory (1965).